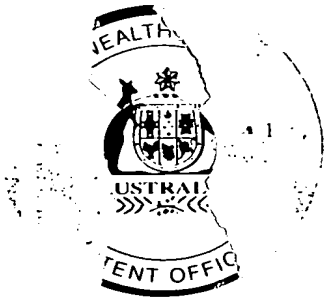
09/392 550

Patent Office
Canberra

I, KIM MARSHALL, MANAGER EXAMINATION SUPPORT AND SALES hereby certify that annexed is a true copy of the Provisional specification in connection with Application No. PQ 0282 for a patent by CANON KABUSHIKI KAISHA filed on 10 May 1999.



WITNESS my hand this
Thirteenth day of September 1999

KIM MARSHALL
MANAGER EXAMINATION SUPPORT
AND SALES

S & F Ref: 461676

ORIGINAL

AUSTRALIA

Patents Act 1990

PROVISIONAL SPECIFICATION FOR THE INVENTION ENTITLED:

Hyperprint Method and Apparatus

Name and Address of Applicant: Canon Kabushiki Kaisha, incorporated in Japan, of 30-2, Shimomaruko 3-chome, Ohta-ku, Tokyo, 146, JAPAN

Name of Inventor: Ernest Yiu Cherong Wan

This invention is best described in the following statement:

HYPERPRINT METHOD AND APPARATUS

Field of the Invention

The present invention relates to preparing hypermedia documents that retain their hyperlinks in the printed copy, and more particularly to an authoring system for providing a near-optimal assignment of physical links to hyperlinks.

Background

Hypermedia documents are computer-based electronic documents that contain text, graphics, audio and video on pages that are connected by navigational links. The navigational links, often referred to as the hyperlinks, permit non-sequential or non-linear traversal of the document by the readers. A well-known source of hypermedia documents is the so-called World Wide Web (WWW) or simply, "the Web".

Hypermedia documents allow multiple simultaneous views and efficient non-linear exploration of information that are not possible with conventional printed documents such as books. On the other hand, unlike books, the absence of an obvious linear structure and a sense of physical orientation in hypermedia documents allows users to become easily lost in the hyperspace of the document.

Although hypermedia documents may be printed, the hyperlinking functionality is typically lost in the printed copy. Most hypermedia documents, especially those on the Web are intended for viewing on the screen and designed to exploit the hyperlinking functionality. As a result, readability also suffers with the loss of the hyperlinks. For instance, removing the hyperlink to the definition of an unfamiliar term may make a description unclear to the readers.

Australian Patent Publication No. AU-A-83194/98 (Attorney Ref: 432345 CFP0954AU MMedia02), corresponding to United States of America Patent Application No. 09/148,475 discloses a method for forming hypermedia documents that retain their electronic hyperlinks as physical hyperlinks within the printed reproduction of the document. The method associates the hyperlinks with cut-outs or tabs on the edges of the pages and lines or other indicia printed onto the page. To traverse a

hyperlink, the reader follows the indicia, places a finger or thumb within the cut-out, locates the first uncut page surface below the cut-out and opens that page.

As the number of hyperlinks increase, it is apparent that the number of cut-outs and thus hyperlinks, is limited by the physical dimensions of the pages. It is therefore desirable to reuse the cut-outs for multiple hyperlinks and optimise their assignment.

Summary of the Invention

In accordance with the present invention there is disclosed a method of creating a document suitable for hard copy reproduction, said method comprising the steps of:

(a) receiving information from at least one source, said information including a plurality of referential links establishing corresponding referential paths between components of said information;

(b) defining a physical structure of said document sufficient to reproduce said information;

(c) defining a plurality of document links corresponding to said referential links;

(d) assigning a user interpretable functional link to each said document link; and

(e) optimising a number of said user interpretable functional links by assigning appropriate plural ones of said document links to individual ones of said functional links.

In accordance with the present invention there is disclosed an authoring system for the creation of a linear document having non-linear referential links, said system including:

means for specifying a linear document structure and the hyperlinks of a hypermedia document;

means for associating said hyperlinks with physical links able to be formed in pages of said document;

means for modelling each said physical link using a one-dimensional vector; and

optimising an assignment of the physical links to one or more of said hyperlinks.

Brief Description of the Drawings

A preferred embodiment of the present invention will now be described with reference to the drawings and Appendix 1 in which:

Fig. 1 is an example of a typical electronic hypermedia document obtained from the World Wide Web;

Fig. 2 shows an electronic hard-copy reproducible hypermedia document constructed from that of Fig. 1 using the method described by the aforementioned Australian Patent Publication;

Fig. 3 shows the printed version of the electronic hypermedia document of Fig. 2;

Fig. 4 is a summary of graphical notations of physical hyperlinks used in the preferred embodiment;

Fig. 5 is a flow diagram of the preferred embodiment of the authoring system that creates hypermedia document templates or hypermedia documents with physical hyperlinks;

Fig. 6 provides an example of a restaurant's guide created using the authoring system of Fig. 5;

Fig. 7 is a diagrammatic representation of the cut-out tabs of the restaurant's guide of Fig. 6;

Figs. 8A and 8B illustrate optimisation of hyperlinks within a page; and

Fig. 9 is a block diagram of a general purpose computer upon which the preferred embodiment of the present invention can be practiced;

Fig. 10A depicts an example of a prior-hypermedia document; and

Fig. 10B depicts a modification of the document of Fig. 10A according to a preferred embodiment where the main knowledge is used to group together hyperlinked information.

Detailed Description

The preferred embodiment of the present invention is described as a computer application program hosted on the Windows™ operating system developed by Microsoft Corporation. However, those skilled in the art will recognise that the described embodiment may can be implemented on computer systems hosted by other operating systems. For example, the preferred embodiment can be performed on computer systems running UNIX™, or OS/2™. The application program has a user interface which includes menu items and controls that respond to mouse and keyboard operations. The application program has the ability to transmit data to one or more printers either directly connected to a host computer or accessed over a network. The application program also has the ability to transmit and receive data to a connected digital communications network (for example the "Internet").

The preferred embodiment of the invention can be practised using a conventional general-purpose (host) computer system, such as the computer system 40 shown in Fig. 9, wherein the application program discussed above and to be described with reference to the other drawings is implemented as software executed on the computer system 40. The computer system 40 comprises a computer module 41, input devices such as a keyboard 42 and mouse 43, output devices including a printer 13 and a display device 11. A Modulator-Demodulator (Modem) transceiver device 52 is used by the computer module 41 for communicating to and from a computer network, for example connectable via a telephone line or other functional medium. The modem 52 can be used to obtain access to the Internet, and other network systems.

The computer module 41 typically includes at least one processor unit 45, a memory unit 46, for example formed from semiconductor random access memory (RAM) and read only memory (ROM), input/output (I/O) interfaces including a video interface 47, and an I/O interface 48 for the keyboard 42 a mouse 43 and optionally a joystick (not illustrated). A storage device 49 is provided and typically includes a hard disk drive 53 and a floppy disk drive 54. A CD-ROM drive 55 is typically provided as a non-volatile source of data. The components 45 to 49 and 53 to 55 of the computer module 41,

typically communicate via an interconnected bus 50 and in a manner which results in a conventional mode of operation of the computer system 40 known to those in the relevant art. Examples of computers on which the embodiments can be practised include IBM-PC/ATs and compatibles, Sun Sparcstations or alike computer systems. Typically, the application program of the preferred embodiment is resident on a hard disk drive 53 and read and controlled using the processor 45. Intermediate storage of the program and the print list and any data fetched from the network may be accomplished using the semiconductor memory 46, possibly in concert with the hard disk drive 53. In some instances, the application program may be supplied to the user encoded on a CD-ROM or floppy disk, or alternatively could be read by the user from the network via the modem device 52.

Referring to Fig. 1, an electronic multi-page hypermedia document 100 is shown which illustrates a typical layout of multi-media content spread over pages 120,122,124 which also include hyperlinks 140,142. It will be appreciated that these are but some of the pages and hyperlinks in document 100. Navigating through the pages is accomplished by traversing the hyperlinks initiated by a signal such as a mouse-click or a finger-touch on the hyperlink anchors. For instance, selecting an anchor 160 of hyperlink 140 will cause page 122 to be displayed, whilst selecting an anchor 162 of hyperlink 142 will cause page 124 to be displayed. The function of the hyperlinks are to be retained in the printed copy of the document.

Fig. 2 shows an equivalent electronic multi-page hard copy reproducible hypermedia document 300 constructed from the document 100 in accordance with the method described in the aforementioned Australian Patent Publication, and which allows the converted document to retain its hyperlinks in the printed copy. The content pages of the document 100 are reformatted to satisfy the physical constraints of the print media. In the example shown, pages 120 and 124 of the document 100 are laid out on the odd-numbered pages 320 and 324 respectively, while page 122 of document 100 is split into an even-numbered page 322 and an odd-numbered page 323. Fig. 3 shows the printed copy 500 of the document 300, wherein the electronics pages 320 and 322 of the

document 300 are printed on the two sides of the physical page 520, and the electronics pages 323 and 324 are printed on the odd-numbered side of the physical pages 523 and 524 respectively.

In this example, the electronic hyperlinks of the document 100 are associated with cut-outs on the edges of the pages. To traverse a hyperlink, the reader places a finger or thumb within the cut-out, locates the first uncut page surface below the cut-out and opens that page. In the document 300, the hyperlinks 140,142 of the document 100 become associated with tabs 340 and 342 respectively which form cut-outs 540 and 542 in the printed copy 500 of Fig. 3 retaining their hyperlinking functionality. On page 320 of the document 300, selecting the tab 340 causes the page 322 to be displayed in a one-page mode and both pages 322 and 323 to be displayed in a two-page mode. Moreover, on page 320, selecting the tab 342 causes page 324 to be displayed in one-page display mode and page 324 and its opposite page to be displayed in a two-page display mode. Note that tab 342 is not active on page 323 as indicated, in this particular implementation, by the absence of a label.

Due to the limited amount of space on the edges of the printed copy, it is possible that some hyperlinks in an electronic hypermedia document cannot be retained in its printed copy as cut-out tabs. While the number of cut-out tabs that can be made along the edges is limited, under certain conditions, it is possible to assign multiple hyperlinks to one cut-out tab. Moreover, related hyperlinks can be grouped to form a set of nested tabs of progressive size along a single axis, for instance, 344 in document 300 and 544 in the corresponding copy 500. When using a set of nested tabs, the user selects which hyperlink to traverse by simply moving the thumb or finger slightly to grasp the desired size of the cut-out.

The preferred embodiment optimises the assignment of the cut-outs and other physical hyperlinks. More specifically, the preferred embodiment provides a mathematical model for describing hyperlinks and cut-out tabs and uses the model to optimise the number of tabs and axes that are required for implementing a given set of hyperlinks. The preferred embodiment will be explained using a particular authoring

system that creates hypermedia documents with physical hyperlinks. It will be appreciated that the present invention can be used in other authoring processes as well as in processes that convert existing electronic hypermedia documents to a format that supports physical hyperlinks.

A physical hyperlink from an anchor (or hot-spot) on page **S** to a target on page **T** can be represented by a data structure comprising of the 1-dimensional vector $\overrightarrow{ST}$ describing the path traversed by the hyperlink and the location of the target page, that is, the page number of **T**. Since hyperlinks are typically implemented physically as cut-out tabs and can be in either a forward or a backward direction, bi-dimensional vectors are involved. The same basic data structure is used to represent the cut-out tabs. Each physical hyperlink in the document is typically associated with one cut-out tab. However, a number of hyperlinks may be able to share one cut-out tab. The preferred embodiment involves a process that starts with a set of cut-out tabs each representing one hyperlink of the hypermedia document. The set of cut-out tabs is then gradually reduced by combining cut-out tabs which can serve multiple hyperlinks.

Other physical attributes such as the size of the tabs and the page, as well as certain system parameters such as the edges to be used for tabs, the maximum number of tabs that can be nested along a single axis and the density of the tabs along an edge are required when optimizing the placement of the cut-out tabs and are typically retained in a tab data structure. Additional attributes such as the location of the tabs on an edge, the shape of the tabs, the colors and labels used by the tabs, etc. are required when producing the tabs. Default values or user's inputs can be used for the value of those attributes that are not set by the optimization process. The author should be allowed to examine the results of the optimization process and make adjustments as required.

A tab running from **S** to **T** has to pass through all the pages in between and is visible, although not necessarily required by those pages. Appropriate presentational style may be used to distinguish an inactive tab from an active tab, for instance by

removing or greying out the label of a tab when it has no associated hyperlink. An array is added to the data structure of the cut-out tab to store the status of the tab for the intermediate pages. In case the tab is shared by multiple hyperlinks, a second array is used to store the labels of the tab for the intermediate pages. Hereafter, depending on the context, the notation **ST** is used to denote a hyperlink or a cut-out tab from page **S** to page **T**. In addition, Fig. 4 shows graphical notations used to depict the cut-out tabs diagrammatically. The graphical notations can also be used in the graphical user interface (GUI) of an authoring system or editing tool to visualize the assignment and the physical arrangement of the tabs.

Fig. 5 shows the preferred embodiment of a system 600 for authoring hypermedia documents with physical hyperlinks. The system 600 optimises the assignment of hyperlinks to cut-out tabs. Fig. 6 shows, in two-page display mode, the representative pages of a hyperlinked restaurant's guide 800 created by the system 600. Fig. 7 shows the diagrammatic representation of the cut-out tabs of the restaurant's guide 800. The restaurant's guide will be used as an example in the description of the system.

Fig. 6 illustrates the starting page of the main sections of the restaurant's guide 800. The sections are defined as indicated in step 610 of Fig. 5 which provides the linear structure of the document 800. Each section is a group of related pages and has an associated identifier. The root (or starting) pages of document 800 is denoted as **R**, followed by the restaurant directories $\mathbf{D}_i$, for m types of cuisines, where $i=1,\ldots,m$. The restaurants are also grouped into sections related to the suburb in which each is located, one for each of the n listed suburbs. Each suburb section has an overview page $\mathbf{S}_j$ and m cuisine sub-sections $\mathbf{C}_{ij}$, $i=1,\ldots,m$, one for each of the m cuisines where $1 \leq j \leq n$. In summary, the document 800 has the (linear) structure:

$$\mathbf{R}\ \mathbf{D}_1\ \mathbf{D}_2\ \ldots\ \mathbf{D}_m\ \mathbf{S}_1\ \mathbf{C}_{11}\ \mathbf{C}_{21}\ \ldots\ \mathbf{C}_{m1}\ \ldots\ \mathbf{S}_n\ \mathbf{C}_{1n}\ \mathbf{C}_{2n}\ \ldots\ \mathbf{C}_{mn}$$

To allow document 800 to be traversed in a non-linear manner among the various sections, a number of hyperlinks are defined in step 612. Using the graphical

notations of Fig. 4, the hyperlinks are denoted as indicated in Table 1. In step 614, each of these hyperlinks is assigned a tab.

TABLE 1

Hyperlinks	Description
$\mathbf{RD}_i$, $i = 1,\ldots,m$	from the root page to each cuisine directory
$\mathbf{D}_r\mathbf{D}_s$, $r,s = 1,\ldots,m$, $r \neq s$	from each cuisine directory to another cuisine directory
$\mathbf{D}_i\mathbf{S}_j$, $i = 1,\ldots,m$, $j = 1,\ldots,n$	from each cuisine directory to each suburb section
$\mathbf{S}_j\mathbf{D}_i$, $i = 1,\ldots,m$, $j = 1,\ldots,n$	from each suburb section to each cuisine directory
$\mathbf{RS}_j$, $j = 1,\ldots,n$	from the root page to each suburb section
$\mathbf{S}_j\mathbf{C}_{ij}$, $i = 1,\ldots,m$, $j = 1,\ldots,n$	from each suburb section to each of its local cuisine sections
$\mathbf{C}_{rj}\mathbf{C}_{sj}$, $r,s = 1,\ldots,m$, $r \neq s$, $j = 1,\ldots,n$	from each cuisine section of a suburb to the other cuisine sections of the same suburb
$\mathbf{C}_{rj}\mathbf{D}_i$, $r,i = 1,\ldots,m$, $j = 1,\ldots,n$	from each cuisine section of a suburb to the cuisine directories

Note that hyperlink definitions can be either "static" or "dynamic". $\mathbf{RD}_i$ and $\mathbf{C}_{rj}\mathbf{C}_{sj}$ are two examples of a "dynamic" hyperlink definition. In the case of $\mathbf{RD}_i$, a hyperlink is generated between the "static" document element **R** and every occurrence of the "dynamic" document element **D**. In the case of $\mathbf{C}_{rj}\mathbf{C}_{sj}$, a hyperlink is generated between every occurrence of two "dynamic" document elements. In contrast, a "static" hyperlink definition defines a fixed hyperlink between two "static" document elements.

In step 616, the number of tabs is reduced by assigning multiple hyperlinks to some of the tabs. A tab can be used for multiple hyperlinks that ended at the same page as the tab. In fact, a tab **AB** can potentially be used for all hyperlinks starting at section **A** or at a section between **A** and **B** and ended at section **B** provided that the tab is

properly labelled on all the pages it passes through and that such use is not confusing to the reader. For instance, tab 910 of Fig. 7 running from **R** to $\mathbf{D}_m$ is shared by hyperlinks $\mathbf{RD}_m$ and $\mathbf{D}_p\mathbf{D}_m$ where p=1,...m-1. In addition, a tab leading from a page **A** to a page **B** provides an implicit return path from page **B** to page **A**. Hence, the number of tabs can be reduced by merging tab pairs that have their two tabs start on the page at which other tab ends.

In step 618, related tabs are nested to reduce the total number of axes required. Nesting tabs on the same axis allows hyperlinks with different target pages to share the same axis therefore making available edge space for other tabs. However, the overall size of the resulting set of nested tabs is larger than a normal tab. Hence, the saving is not exactly *n* fold for nesting *n* tabs. Moreover, indiscriminate nesting of tabs may introduce artificial grouping of otherwise unrelated content and confuse the reader. Hence, a set of nested tabs is preferably used only where a natural or logical relationship exists among the linked materials. Such relationship may be inferred from the structure of the documents. For instance, the set of nested tabs 912 of Fig. 7 are used for assessing the *m* restaurant directories each for a different cuisine. In this case, the nesting of tabs reduces the number of axes required from *m* to 1.

Steps 616 and 618 can be fully-automated. However, an editing tool that allows the user to adjust the resulting hyperlinks/cut-out tabs assignment interactively is sometimes desirable.

In step 620, presentational style such as the location, colour, size, shape, etc. of the tabs are defined along with the presentational style of the other elements of the documents such as those of the headers, paragraphs, tables, lists, etc. A number of presentational styles 644 can be defined to allow the document to be presented in different ways.

The structural definitions 642 of the hypermedia document created as a result of steps 610 to 618 and the style definitions 644 created as a result of step 620 can be saved and used as a document template 640 for creating documents of the same type in the future. In the case of the restaurant's guide example of Fig. 6, XML and XSL are

used for defining the structure and the style of the document respectively while the hypermedia document 646 is generated in HTML format.

In step 622, content is added either interactively using an editor or from data files using a loader or both. Additional hyperlinks and tabs are generated from the "dynamic" hyperlink definitions. It will be appreciated that all copies of the tabs generated from a "dynamic" hyperlink definition can use the same set of axes. For instance, in Fig. 7, the $\mathbf{RD}_i$'s use a single set of nested tabs 912 and the $\mathbf{C}_{rj}\mathbf{C}_{sj}$'s use the same two axes for their nested tabs 914.

After the content is added, new content specific hyperlinks and their associated tabs can be defined if required as indicated in step 624. The number of new tabs can also be reduced using the same methods used in step 616 and 618.

It will be appreciated that the hyperlinks in the resulting hypermedia document 646 are optimally or near-optimally assigned to the cut-out tabs that serve as physical hyperlinks in the printed copy.

Although the forgoing embodiment describes optimising the hyperlinks spread across a number of page using cut-out tabs, the same optimising principles may be applied to individual pages and links on those pages. Fig. 8A shows a traditional arrangement of a hyperlinked page 1000 which includes four hyperlink anchors (ANCHORS 1-4) and two hyperlink destinations, one being a cut-out tab 1002 formed in the page 1000, and the other being a LINKED_COMPONENT printed on the page 1000. As seen, printed hyperlinks 1004 and 1006 (printed lines on the page) directly connect the ANCHOR 1 and ANCHOR 3 respectively to the LINKED _COMPONENT. Similarly, hyperlinks 1008 and 1010 directly connect ANCHOR 2 and ANCHOR 4 respectively to the cut-out tab 1002. It is apparent from Fig. 8A that some amount of optimisation may be made in order to maximise the available space on the page 1000.

Fig. 8B shows how the page 1000 may be optimised according to another embodiment. As with the multi-page cut-out cases described above, here hyperlinks that point to the same destination are combined as best possible within the confines of the document to minimise the number of links on the page. As seen in Fig 8B, the

printed hyperlinks are altered in their placement on the page with printed hyperlinks 1112 and 1114 from ANCHORS 1 and 3 respectively merging into a hyperlink 1116 which accesses LINKED_COMPONENT. Similarly, hyperlinks 1118 and 1120 from ANCHORS 2 and 4 respectively, merge into a hyperlink 1122 which access the cut-out 1002. In this fashion it is seen that the application of the optimisation process can reduce the space used by the hyperlinks internally referencing a single page.

Whilst the example of Fig. 8B demonstrated the use of printed lines as hyperlink indicia, other indicia may be used. For example, graphic symbols and the like may substitute for the printed lines to provide to the reader the same referential link that otherwise would have been provided in an electronically linked document.

According to the various principles of the embodiments described above, a further embodiment makes use of a knowledge-based subsystem configured to interpret the main knowledge associated with each hyperlink in order to decide if and how certain hyperlinks can be grouped together thus permitting the grouped hyperlinks to be assigned to a set of nested tabs within a hyperlinked documents. Whilst it is possible for grouping of information within the document to be based upon the inherent structure of the electronic document from which the hyperprint physical document is to be derived, such is limited to the extent of structure incorporated in the electronic document and requires the author of the electronic document to specify every grouping interactively. This can be a tedious process and is not a practical solution where the principles of the present invention are intended to provide for the automated translation of a hyperlinked electronic document to a hyperlinked text document.

Figs. 10A and 10B illustrate such an arrangement in which Fig. 10A depicts an example hypertext document incorporating physical hyperlinks according to the disclosure of the aforementioned Australian Patent Publication. As seen, the document relates to a motor car and the pages displayed in Fig. 10A depict various features associated with the motor car including trip computer, air bags, seat belts, climate control, sound system, ABS brakes, anti-submarining seats, power steering and four

wheel steering. As seen, each link is physically connected to a corresponding cutout or other physical indicator of the hyperlinked information.

According to the present embodiment and as illustrated in Fig. 10B, the various features shown in Fig. 10A may be grouped according to the particular type of feature depicted. As will be understood by those familiar with motor cars, the various features illustrated in Fig. 10A may be grouped into two categories, those relating to safety features of the motor car, and those relating to the comfort of the user of the motor car. According to the specific embodiment illustrated in Fig. 10B, the various hyperlinks associated with each of the groupings of safety and comfort are combined for the various features indicated on the printed page. As seen, each of the combined hyperlinks refers to a nested arrangement of tabs that may be used in the manner described above to access information relating to the specific features in each grouping.

The embodiment of Fig. 10B relies upon the main knowledge associated with each of the hyperlinks, such knowledge including information regarding the overall grouping for either safety or comfort. A grouping can be used to select or recommend specific graphic symbols, colours, or abbreviated labels and the like associated with any of the cutout tabs and/or indicia printed upon the page.

In a further embodiment, style sheets and/or style definitions may be used, not only in specifying the presentational style of cutout tabs used in forming the hyperlinked document, but also in specifying those cutout tabs that are to be implemented for a particular view of the document.

Existing web technology allows the use of XSL-style sheets (XSL = Extensible Stylesheet Language) to extract various components of XML documents for presentation thereby allowing the presentation of different views of the document. Similarly, and according to the present embodiment, different style sheets may be used in selecting those hyperlinks that are important in generating various views of the hyperprint document. In order to achieve such different views, the hyperlinks are required to be marked up according to their particular role, purpose and level of importance.

For example, such a configuration permits the links and tabs, which form the essence of the hypertext document, to be manipulable (editable) like data in any document. For example, with reference to Fig. 10B, a particular style sheet may provide for the printed hyperlinks (eg. a directional arrow), to be provided in a variety of colours or in some other form such as formed using dashed or dotted lines. A modified style sheet may also provide for alternate shaped cutout portions such as semi-circular, square, triangular and the like. Style sheets may also be used for modifying any printed colours or patterns associated with nested cutouts and tabs. Further, and with specific reference to the example of Fig. 10B, a specific style sheet may be used to either include or exclude particular types of information to which hyperlinks may apply. That is, Stylesheets can be used to select the set of cut-out tabs that should be included in various versions of the document. For example, where desired, those motor car features of Fig. 10B relating to comfort may be excluded in the generation of the hypertext printed document from the electronically hyperlinked source.

Annexed hereto and marked as Appendix 1 is a copy of yet unpublished paper co-authored by the present inventor and entitled "Retaining Hyperlinks in Printed Hypermedia Document" which provides further discussion on the various features and aspects of the present invention and the embodiments described above.

The foregoing describes only a particular embodiment of the present invention, and modifications and alternatives obvious to those skilled in the art can be practiced within the spirit of the invention. Specific aspects of the invention may now be recited.

Aspects of the Invention

1. A method of creating a document suitable for hard copy reproduction, said method comprising the steps of:

(a) receiving information from at least one source, said information including a plurality of referential links establishing corresponding referential paths between components of said information;

(b) defining a physical structure of said document sufficient to reproduce said information;

(c) defining a plurality of document links corresponding to said referential links;

(d) assigning a user interpretable functional link to each said document link; and

(e) optimising a number of said user interpretable functional links by assigning appropriate plural ones of said document links to individual ones of said functional links.

1A. A method according paragraph 1 wherein step (e) includes the step of importing said information into said structure to form said document, and applying said optimised links to said document.

2. A method according to paragraph 1 or 1A, wherein said physical structure includes a single page and said functional links include indicia printable onto said page, and appropriate plural ones of said document links are merged to form a single said indicia associated with a component on said page.

3. A method according to paragraph 1, 1A or 2 wherein said physical structure includes plural pages and said functional links include cut-out tabs formed in at least one

of said pages, and appropriate plural ones of said document links refer to a single said cut-out tab.

4. A method according to paragraph 3, wherein at least one said cut-out tab is formed as part of a nest of correspondingly located tabs distributed amongst consecutive ones of said pages.

5. A method according to paragraph 4 further comprising the step of retaining structure definitions of said document in a template for formatting subsequent documents in a corresponding fashion.

6. A method according to paragraph 4 or 5 further comprising defining a presentational style to said document and applying said presentational style to said functional links to distinguish said functional links from said components.

7. A method according to paragraph 6 further comprising the step of retaining said presentational style of said document in a template for formatting subsequent documents with a corresponding presentational style.

8. A method according to paragraph 1 further comprising defining content specific document links and incorporating corresponding functional links into said document.

9. A method according to paragraph 8 wherein said content specific document links are user defined.

10. A method according to paragraph 1, wherein step (e) comprises sadi document links are grouped according to predetermined criteria associated with said links, each said group having associated therewith a corresponding at least one optimised link.

11. A method according to paragraph 1, further comprising the step of associating a selected stylistic layout with said optimised functional links so as to vary a hard-copy reproduction of said document.

12. Apparatus for performing the method of any one of the preceding paragraphs.

13. A computer program product incorporating a computer readable medium having a series of program instructions configured to perform the method of any one of paragraphs 1 to 11.

14. An authoring system for the creation of a linear document having non-linear referential links, said system including:

means for specifying a linear document structure and the hyperlinks of a hypermedia document;

means for associating said hyperlinks with physical links able to be formed in pages of said document;

means for modelling each said physical link using a one-dimensional vector; and

optimising an assignment of the physical links to one or more of said hyperlinks.

15. A method of creating a document substantially as described herein with reference to any one of the embodiments as illustrated in the drawings.

16. A document creation system substantially as described herein with reference to any one of the embodiments as illustrated in the drawings.

Dated 10 May, 1999
Canon Kabushiki Kaisha
Patent Attorneys for the Applicant/Nominated Person
SPRUSON & FERGUSON

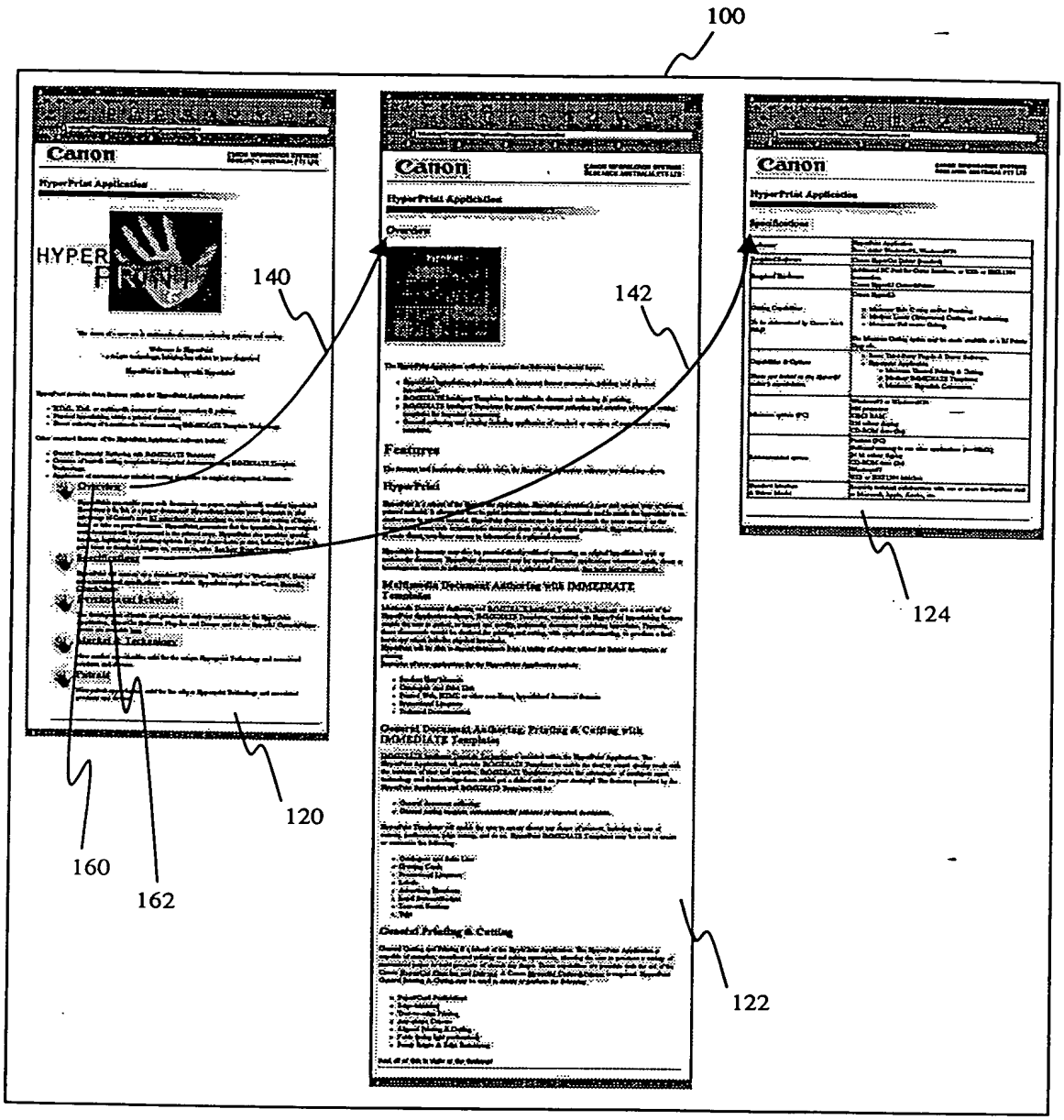


Figure 1

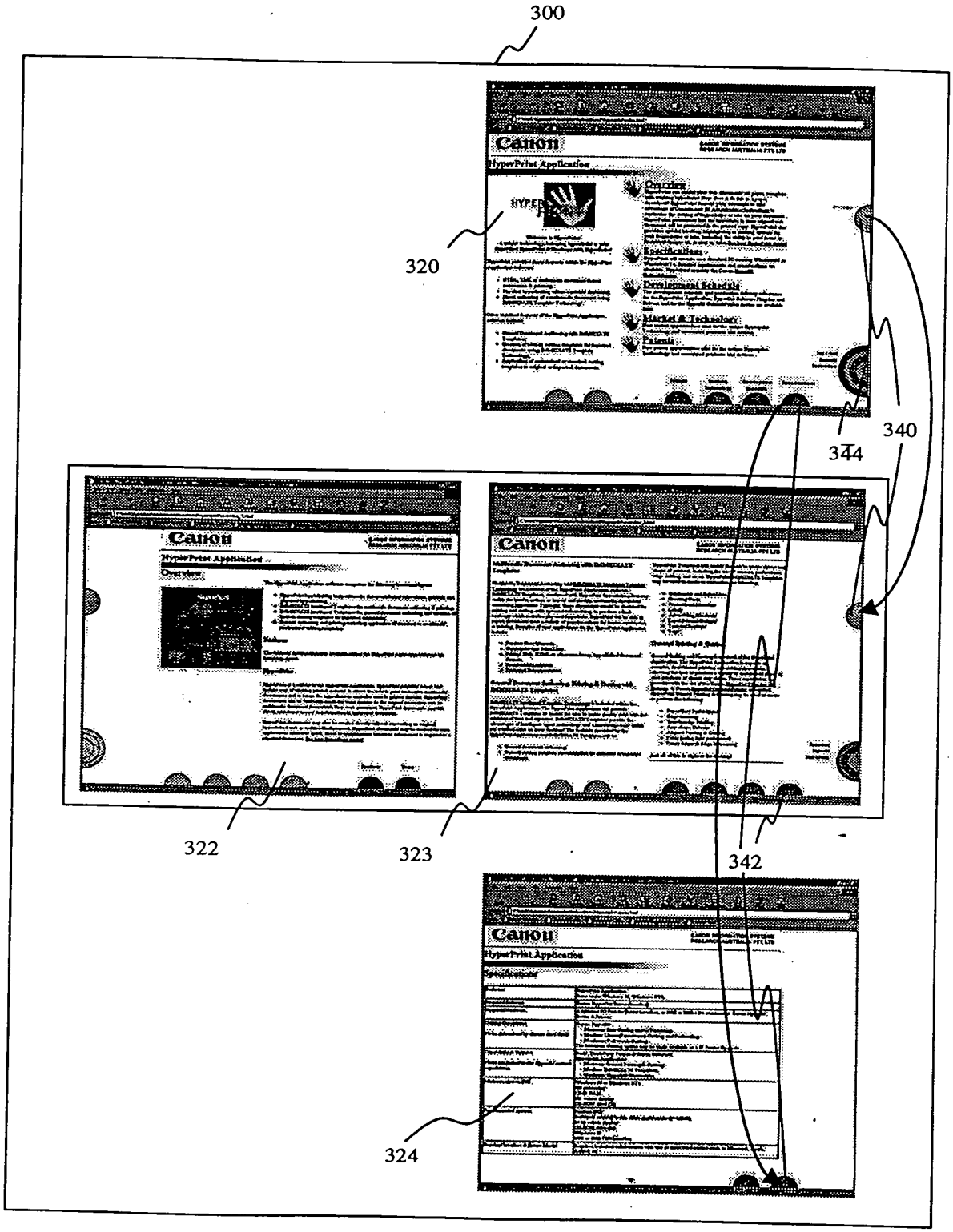


Figure 2

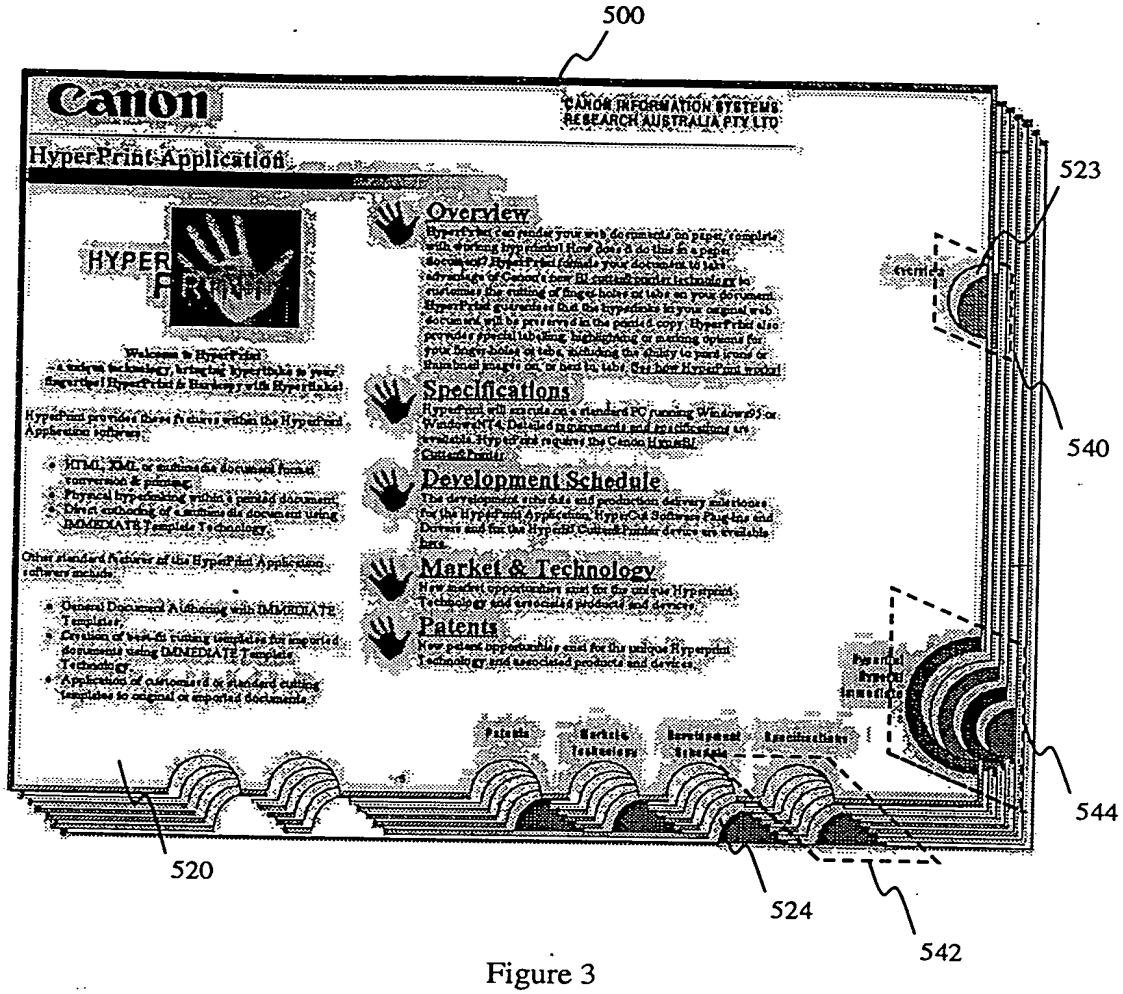

500
Canon
CANON INFORMATION SYSTEMS RESEARCH AUSTRALIA PTY LTD
HyperPrint Application
Overview
Specifications
Development Schedule
Market & Technology
Patents
523
540
544
520
524
542

Figure 3

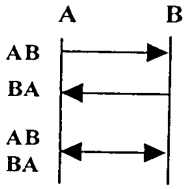
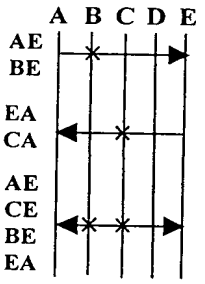
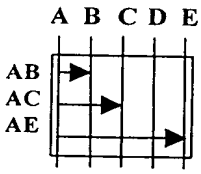
	**Section:** A vertical bar denotes a section, that is, a set of one or more consecutive pages. The identifiers above the vertical bar denote the end-points of the hyperlinks that start or end at the section, that is, the anchors and targets defined within the section.
	**Tab:** A cut-out tab is denoted by an arrow. A left pointing arrow represents a forward link and a right pointing arrow represents a backward link. An arrow with arrow head at both ends represents a bi-directional tab that provides both a forward and a backward link. The arrow is labelled. The label indicates the hyperlinks that are associated with the tab. The label has the format **ST** where **S** is the identifier of the anchor and **T** is the identifier of the target.
	**Shared tabs:** To denote a cut-out tab that is shared by a set of hyperlinks, a cross is placed at the anchor section of each hyperlink except the outermost one along the arrow that represents the tab.
	**Nested tabs:** A set of arrows enclosed in a rectangular box denotes a set of nested cut-out tabs.

Figure. 4

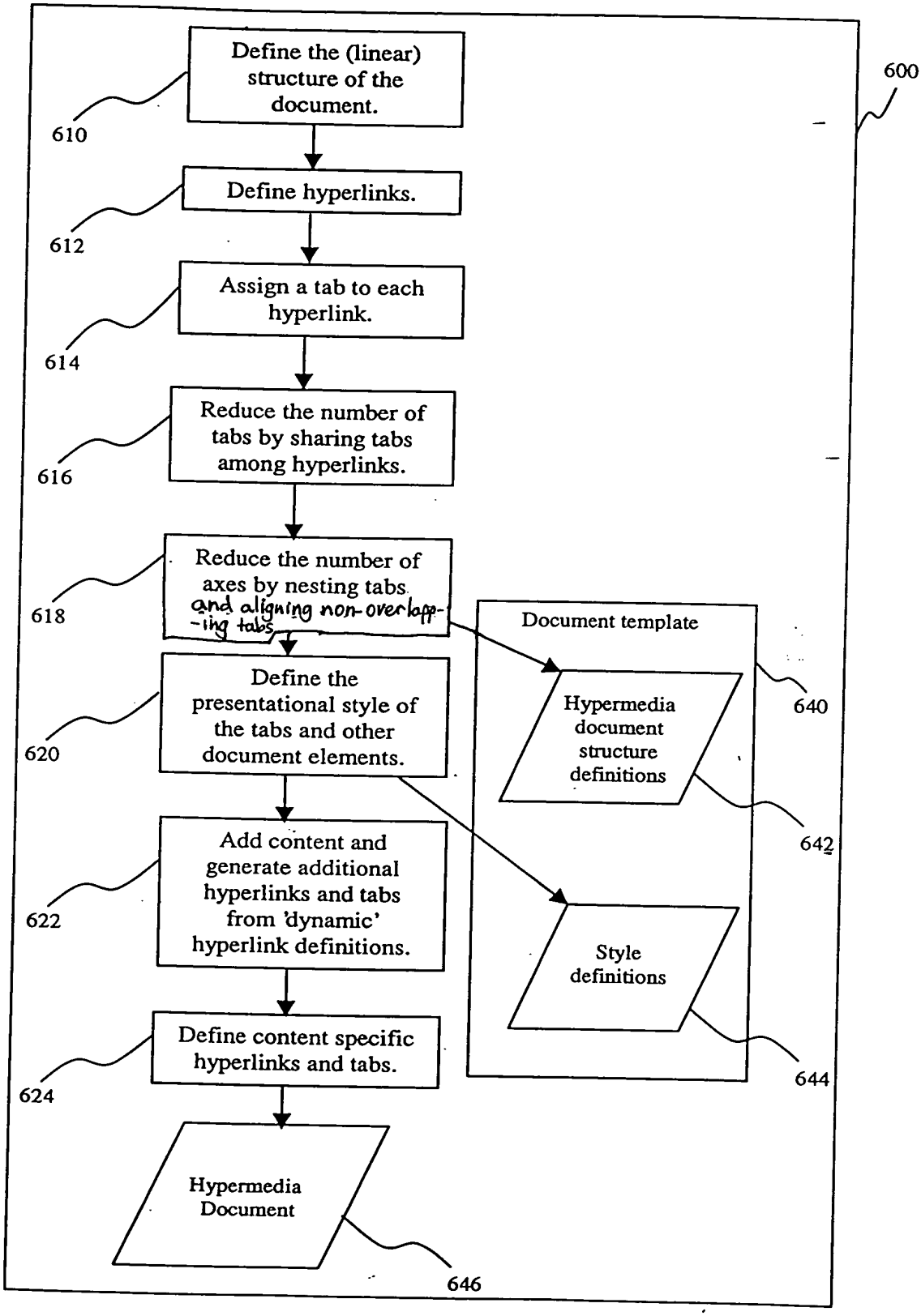

600
Define the (linear) structure of the document.
610
Define hyperlinks.
612
Assign a tab to each hyperlink.
614
Reduce the number of tabs by sharing tabs among hyperlinks.
616
Reduce the number of axes by nesting tabs and aligning non-overlapping tabs
618
Define the presentational style of the tabs and other document elements.
620
Add content and generate additional hyperlinks and tabs from 'dynamic' hyperlink definitions.
622
Define content specific hyperlinks and tabs.
624
Hypermedia Document
646
Document template
640
Hypermedia document structure definitions
642
Style definitions
644

Figure 5

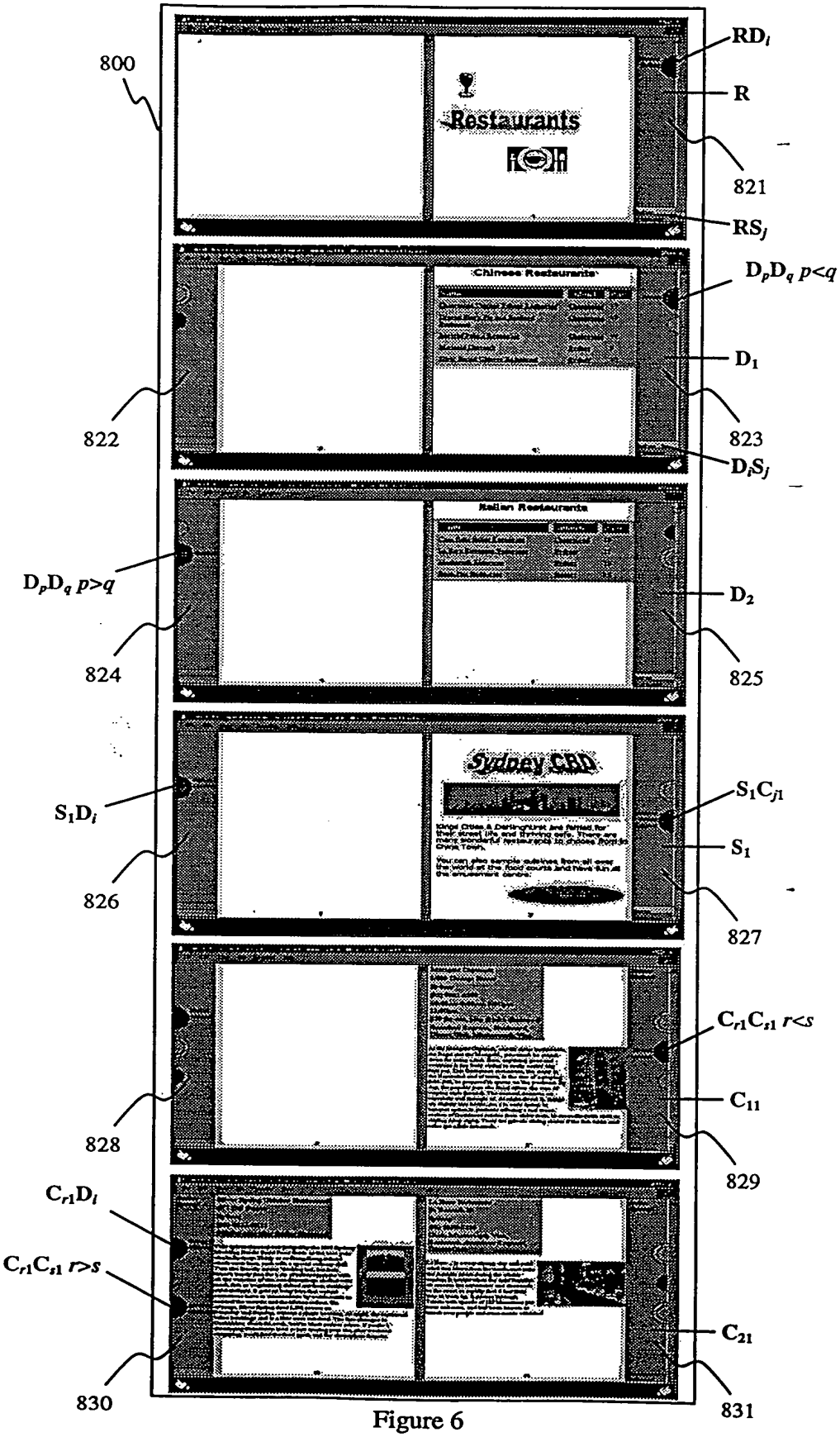

800
RD$_i$
R
821
RS$_j$
D$_p$D$_q$ $p<q$
D$_1$
822
823
D$_i$S$_j$
D$_p$D$_q$ $p>q$
D$_2$
824
825
S$_1$D$_i$
S$_1$C$_{j1}$
S$_1$
826
827
C$_{r1}$C$_{s1}$ $r<s$
C$_{11}$
828
829
C$_{r1}$D$_i$
C$_{r1}$C$_{s1}$ $r>s$
C$_{21}$
830
831

Figure 6

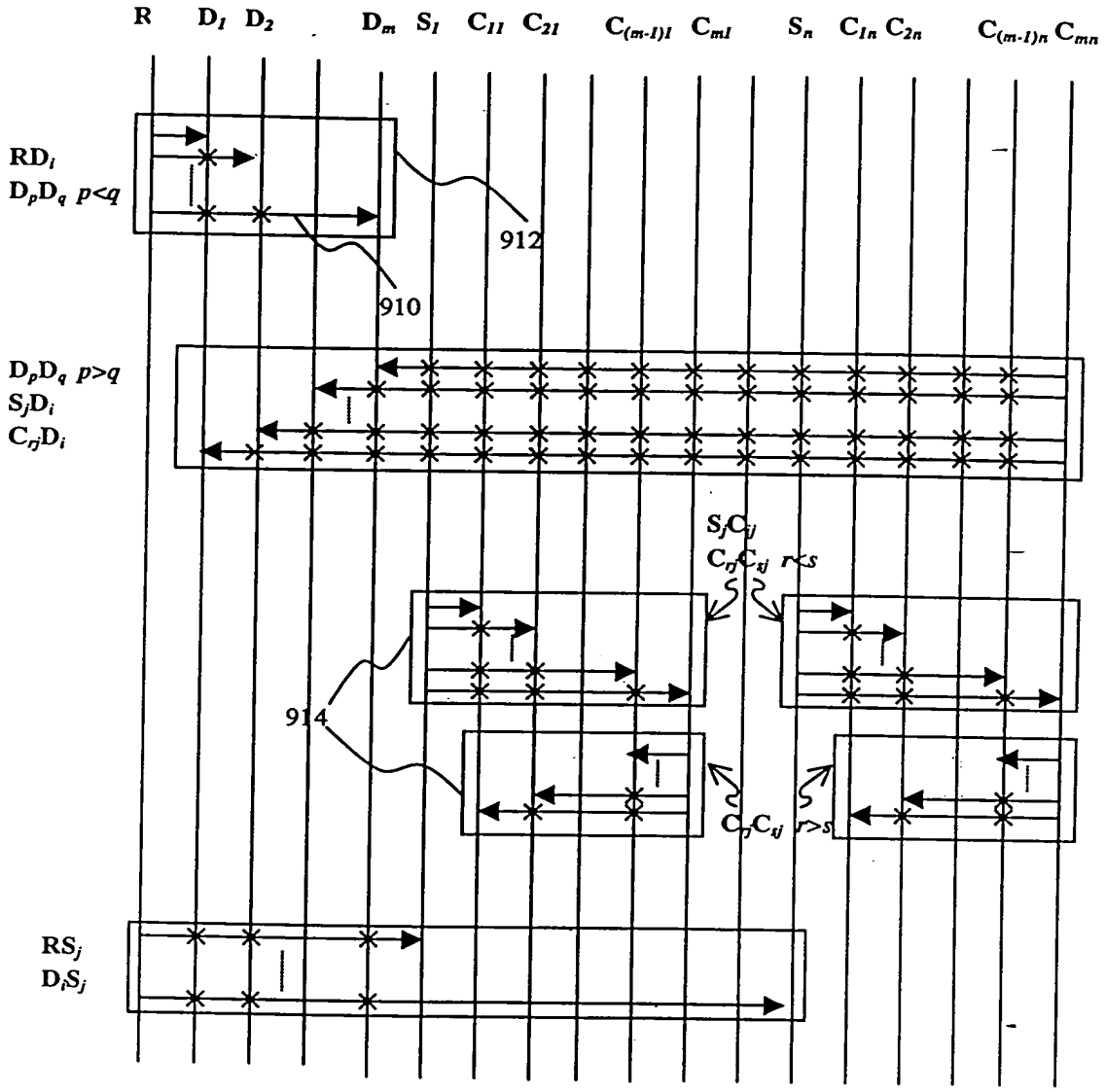

R
D1 D2
Dm S1 C11 C21
C(m-1)1 Cm1
Sn C1n C2n
C(m-1)n Cmn
RDi
DpDq p<q
912
910
DpDq p>q
SjDi
CrjDi
SjCij
CrjCsj r<s
914
CrjCsj r>s
RSj
DiSj

Figure 7

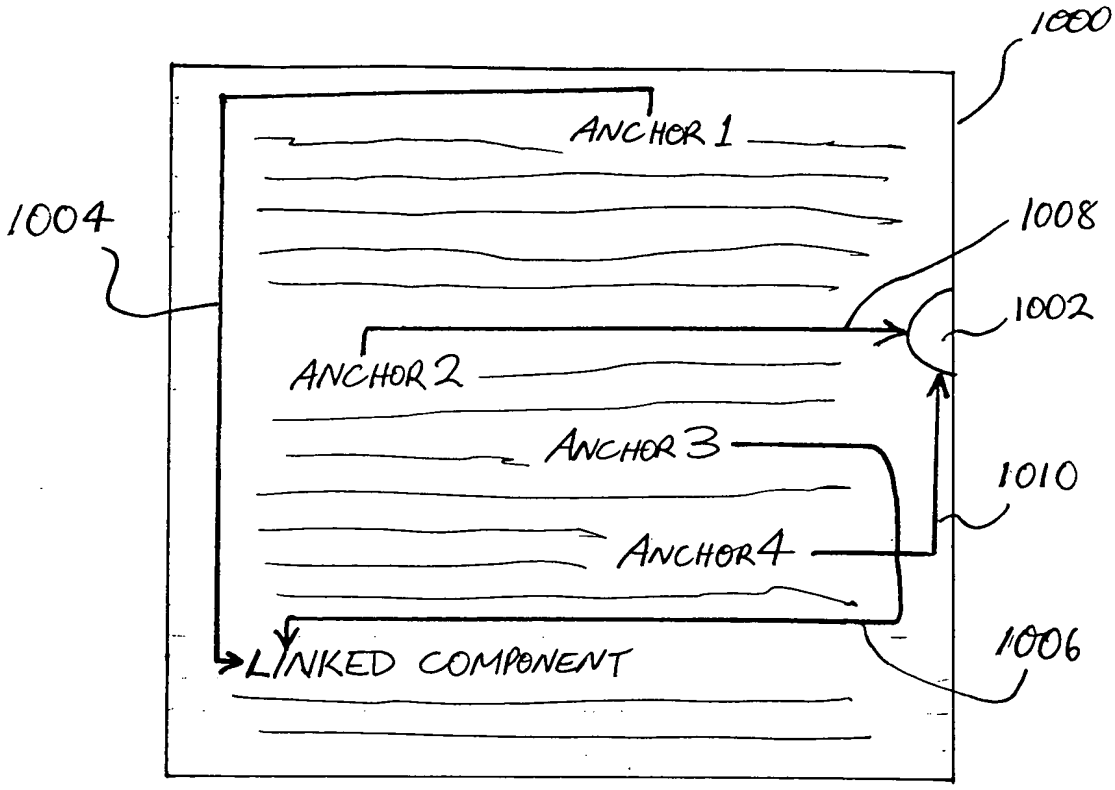

1000
ANCHOR 1
1004
1008
1002
ANCHOR 2
ANCHOR 3
1010
ANCHOR 4
1006
LINKED COMPONENT

FIG. 8A

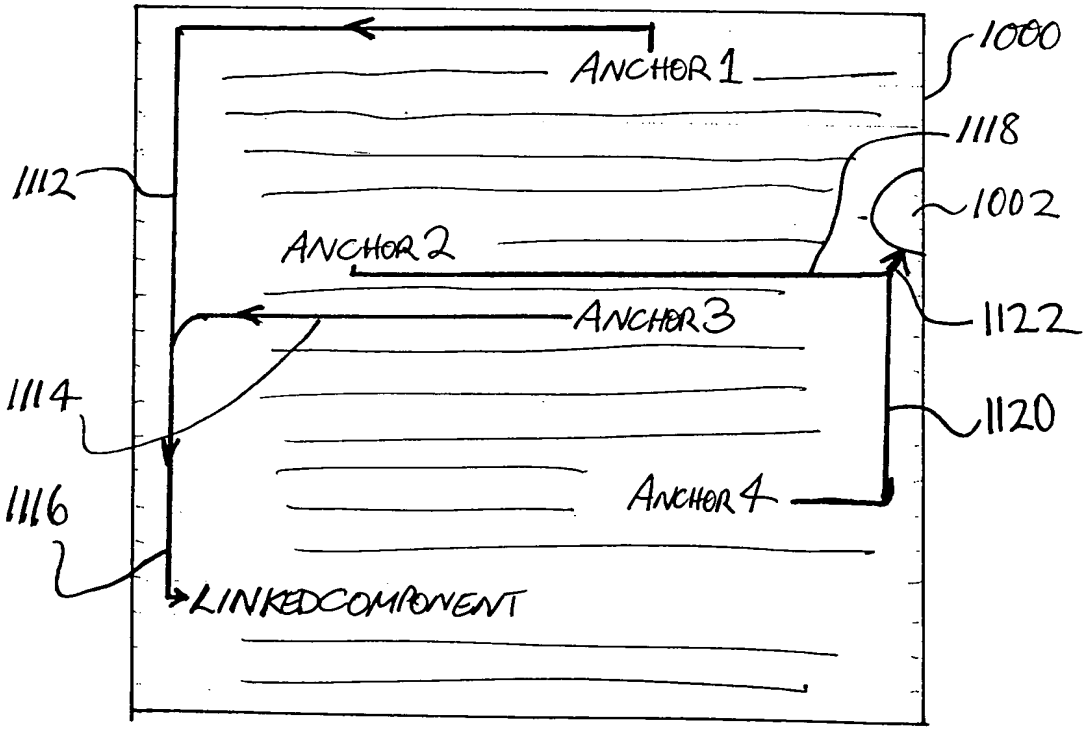

1000
ANCHOR 1
1118
1112
1002
ANCHOR 2
ANCHOR 3
1122
1114
1120
ANCHOR 4
1116
LINKEDCOMPONENT

FIG. 8B

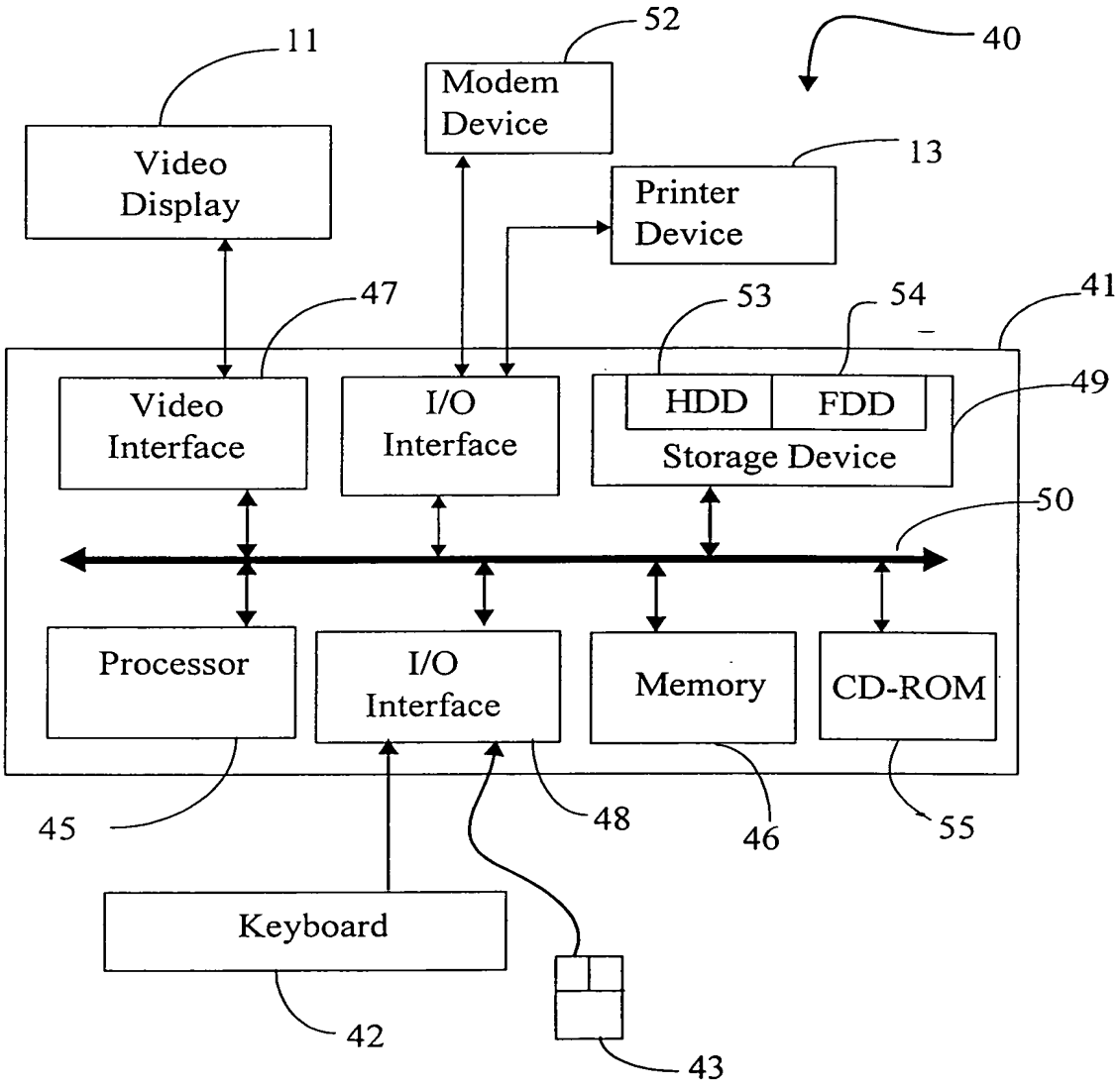

11
52
40
Modem Device
Video Display
13
Printer Device
47
53
54
41
Video Interface
I/O Interface
HDD
FDD
49
Storage Device
50
Processor
I/O Interface
Memory
CD-ROM
45
48
46
55
Keyboard
42
43

FIG. 9

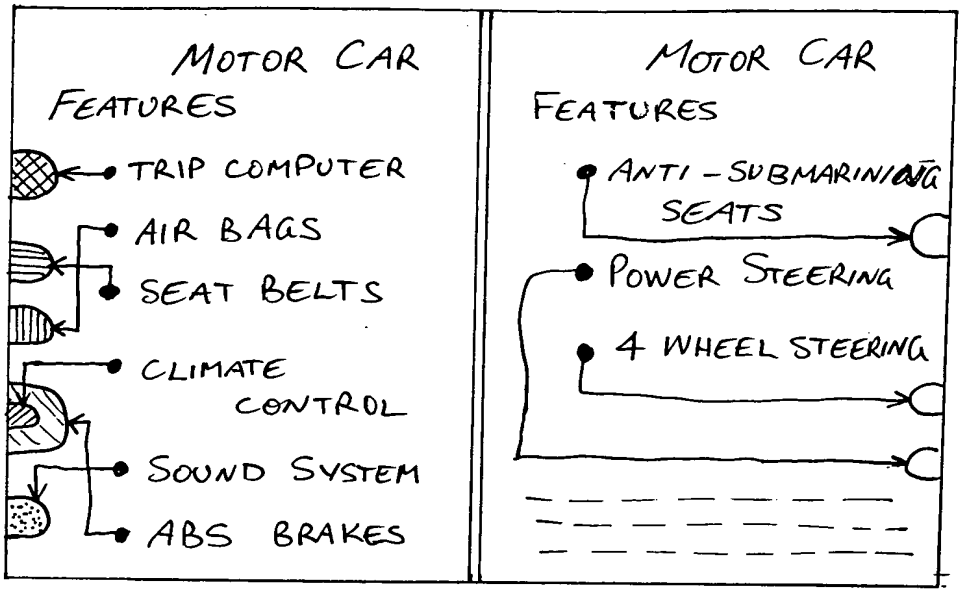


FIG. 10A (PRIOR ART)

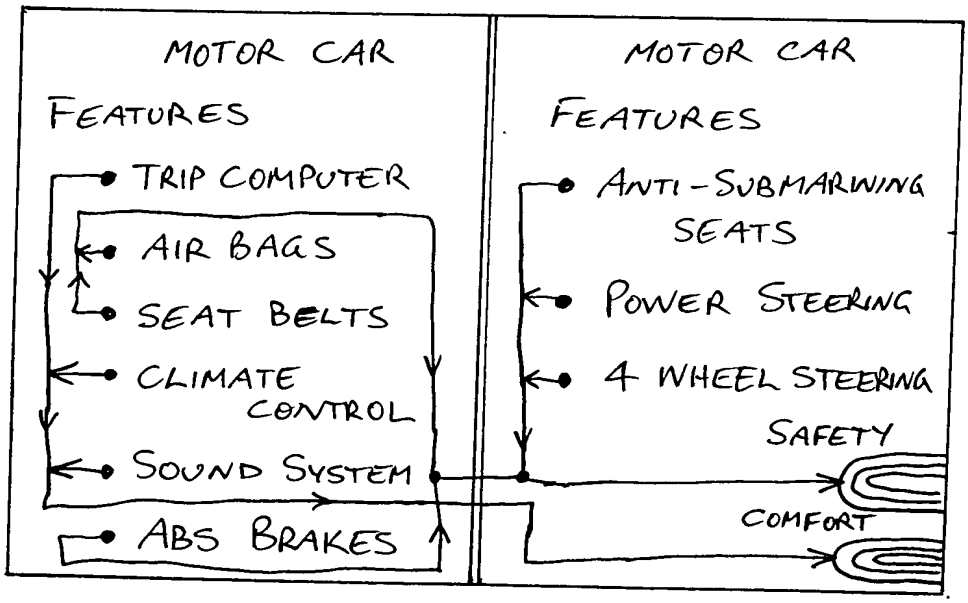


FIG. 10B

APPENDIX 1

Retaining Hyperlinks in Printed Hypermedia Document

Ernest Wan, Philip Robertson, John Brook, Stephen Bruce, and Kristine Armitage

Canon Information Systems Research Australia
1 Thomas Holt Drive, North Ryde
NSW 2113 Australia

ernest@research.canon.com.au, phil@research.canon.com.au, johnb@research.canon.com.au, steveb@research.canon.com.au, and kristine@research.canon.com.au

Abstract

In this paper, we describe a method that allows a hypermedia document to retain its hyperlinks in the printed copy. The method associates the hyperlinks with cut-out tabs on the edges of the printed pages. A method for modelling the cut-out tabs and optimising their assignment to the hyperlinks is discussed. We also describe a prototype authoring system that implements the method.

Keywords

Hyperlink, hypermedia document, cut-out tab, printing, cutting.

1. Introduction

Hypermedia documents are computer-based documents that contain text, graphics, audio and video on pages that are connected by navigational links. The navigational links, often referred to as the hyperlinks, permit non-sequential or non-linear traversal of the document by the readers. A well-known source of hypermedia documents is the so-called World Wide Web (WWW) or simply the Web.

Hypermedia documents allow multiple views and efficient, non-linear exploration of information that are not possible with traditional books. On the other hand, the absence of an obvious linear structure and physical medium make it very easy to get lost in the hyperspace.

Hypermedia documents may be printed. Nevertheless, the hyperlinking functionality is typically lost in the printed copy. Most hypermedia documents especially those on the Web are intended for viewing on the screen and designed to exploit the hyperlinking functionality. As a result, readability also suffers with the loss of the hyperlinks. For instance, removing the hyperlink to the definition of an unfamiliar term may make a description unclear to the readers.

Recently, the World Wide Web Consortium (W3C) has included elements in its Cascading Style Sheet Level 2 (CSS2) specification [2] to allow a Web page to have different styles for different output media. For example, different fonts, margins and colors may be specified for screen displaying and printing. Alternatively, a separate document especially designed for printing may be maintained. In this case, readability of the printed copy can be improved by re-organizing those parts of the documents that are affected by the loss of hyperlinks. Currently, no technique is available for preserving hyperlinks and, hence, the ability of non-linear browsing in printed copy.

Waters *et al* have commented that "the things that hypertext can do best, such as linking ..... and quickly jumping back and forth among complex anchors and bookmarks, are unavailable in printed media". They considered this a major issue in Web printing, sometimes leaving the authors of Web pages with an unacceptable compromise [4]. Considering the inability to print hyperlinks a forgone conclusion, they suggested that "there is a fundamental dichotomy of usage between hypertext and printed information". Their view is commonly shared by other researchers and users of hypermedia documents which explains the lack of references on the subject. Clearly a method for retaining hyperlinks in printed copy would be significant for the printing of hypermedia documents.

In this paper, we describe a method that allows a hypermedia document to retain at least some of its hyperlinks in the printed copy. The method associates the hyperlinks with cut-out tabs on the edges of the physical pages. Techniques for presenting such association visually are discussed. It is obvious that the number of cut-out tabs and thus hyperlinks is

nited by the physical dimensions of the pages. Methods for preserving edge space are described. We also present a ethod for modelling the cut-out tabs and optimising their assignment to the hyperlinks. An experimental system that implements the approach and works on Web pages, is described. Finally, the potential applications of the technique and some practical issues are discussed.

2. Cut-out tabs as physical hyperlinks

Figure 1 shows two typical examples of cut-out tabs and the terminology we are going to use for describing their various characteristics.

2.1. Traditional use of cut-out tabs

Cut-out tabs are sometimes used in traditional books and documents such as dictionaries and operation manuals to facilitate access to the various sections of the document. Cut-out tabs have also been used in some children books to allow non-sequential traversal of the material. The first application provides essentially an indexing function to the content. The later application although more often done for fun than other more subtle practical purpose is nevertheless hyperlinking. Typically, only a small number of cut-out tabs are used and seldom are they vital in delivering the content.

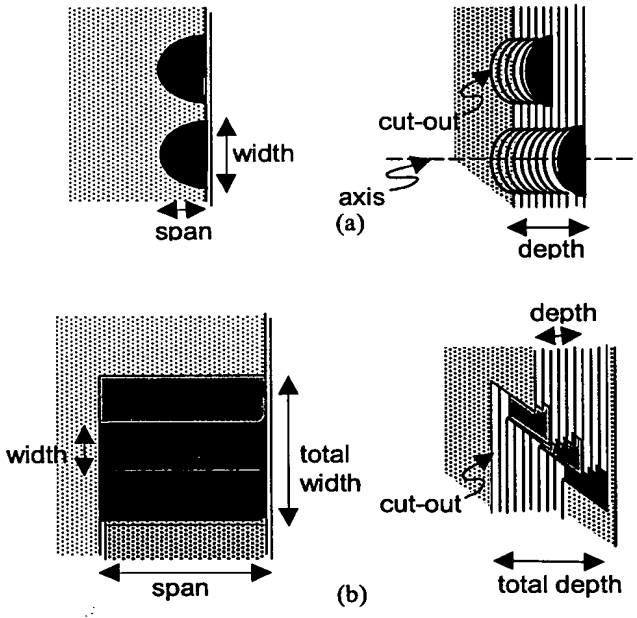


Fig. 1. Examples of cut-out tabs.

2.2. Hyperlink support

Cut-out tabs in paper documents and hyperlinks in hypermedia documents are analogous in both their functionality and appearance. When operating a cut-out tab, the reader places a finger or thumb within the cut-out tab, locates the first uncut page surface below the cut-out tab and opens that page. The action is not unlike clicking a hot-spot or anchor - the start point of a hyperlink - in a hypermedia document and traversing to the target page or the end point of the hyperlink (Figure 2). In addition, like hot-spots, cut-out tabs are visually distinct from their surround.

A cut-out allows traversing the document in both the forward and backward direction. A cut-out tab that implements a forward (backward) link from an earlier page to a later page also provides a implicit backward (forward) path from the later page to the earlier page for free. The implicit return path can be used for hyperlinks linking information in the opposite direction. For instance, the two hyperlinks in Figure 2 can share the same cut-out as shown in Figure 3.

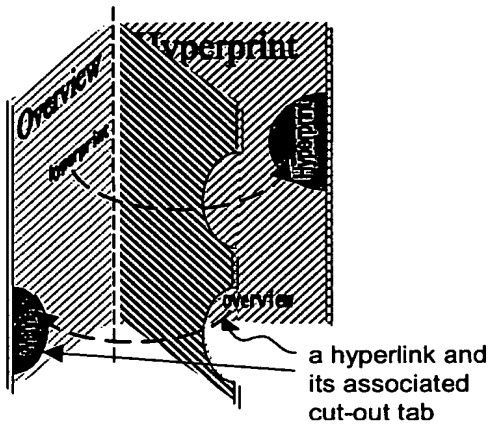


Fig. 2. Traversing hyperlinks in printed hypermedia document.

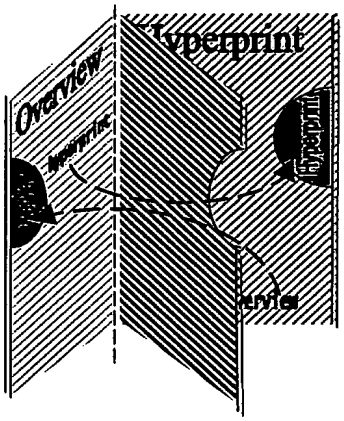


Fig. 3. Utilizing both the forward and backward paths of a cut-out tab for hyperlinks in opposite directions.

2.3. Styles and properties

The cut-out shows part of the linked page. The uncut surface reviewed is usually color-coded or marked with a distinctive pattern or texture. Text label and/or icon can be placed within or alongside the cut-out tab to associate the cut-out tab with a hyperlink (Figure 4).

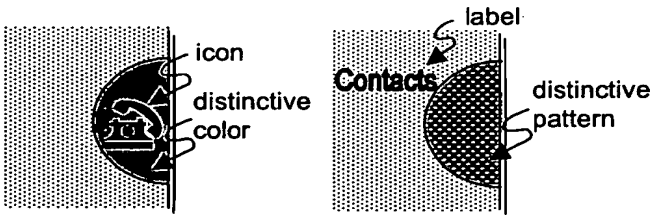


Fig. 4. Styles commonly used for cut-out tabs.

2.3.1. Labelling

A cut-out tab is visible to and has an effect on all the pages the cut-out passes through. Placing the label/icon within a cut-out tab has the implication that the hyperlink is relevant to all the intermediate pages. As an example, this type of labelling is suitable for implementing the "home" hyperlink found in many Web pages.

If the label or icon is placed alongside the cut-out tab, the label/icon can be removed or greyed out for those

ermediate pages where the hyperlink is irrelevant (Figure 5). By placing the label/icon alongside the cut-out tab, it also allows the cut-out tab to be shared by multiple hyperlinks that end at the same target page. Multiple labels/icons can be placed along side the cut-out tab on a single page (Figure 6). In addition, different labels/icons can be used on different pages.

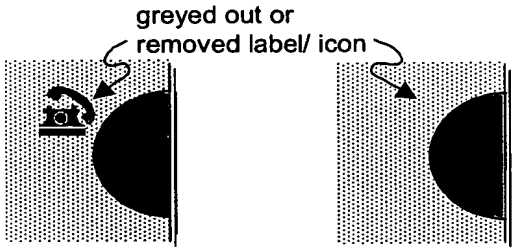


Fig. 5. "Disabled" cut-out tabs on pages where the associated hyperlinks are irrelevant.

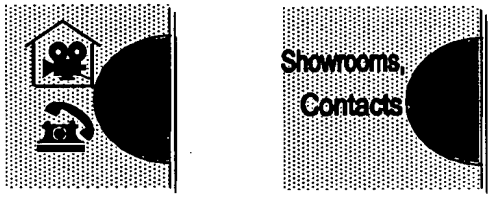


Fig. 6. Multiple hyperlinks that end at the same target page can share a cut-out tab by placing multiple labels/icons alongside the tab.

Instead of placing labels/icons alongside the cut-out tab, text or graphical objects within the content can be used directly. In this case, a connector is used to connect the text/graphical object to the cut-out tab (Figure 7). A coloring scheme or indexing scheme, although sometimes hard to decode when used alone, can also be used to associate the hot-spots to the cut-out tabs (Figure 8).

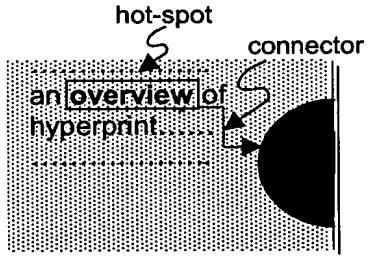


Fig. 7. Using connector to associate a hot-spot with a cut-out tab.

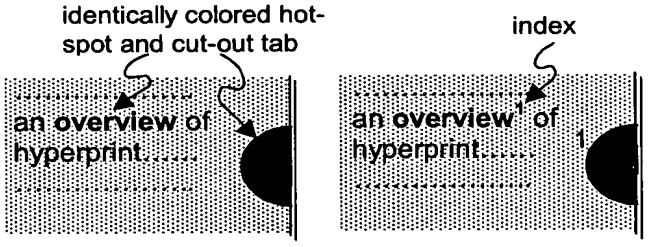


Fig. 8. Using color and index to associate hot-spots with cut-out tabs.

2.3.2. Nested cut-out tabs

Related hyperlinks can be grouped to form a set of nested cut-out tabs of progressive size along a single axis (Figure 9) freeing up edge space for other hyperlinks. When using a set of nested cut-out tabs, the user selects which hyperlink to traverse by simply moving the thumb or finger slightly to grasp the desired size of the cut-out. The set of rectangular tabs shown in Figure 1(b) can be thought of as a special type of nested cut-out tabs.

should be stressed that indiscriminate use of nested cut-out tabs may introduce artificial grouping of otherwise unrelated content and confuse the reader. Hence, nested cut-outs should only be used where a natural or logical relationship exists among the linked materials.

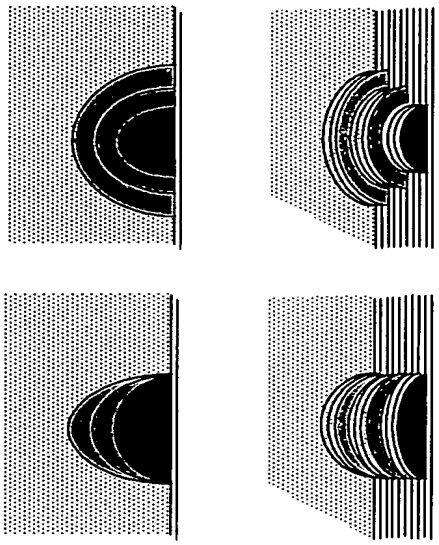


Fig. 9. Nested cut-out tabs.

2.4. Conceptual grouping and perceptual layout

Nesting of cut-out tabs allows related linked materials to be conceptually grouped. Conceptual grouping of a hyperlink and its return path can also be achieved by merging a forward cut-out tab with its corresponding backward (or return) cut-out tab as in Figure 3 and Figure 10. A similar return path can be implemented for the set of rectangular tabs in Figure 1(b).

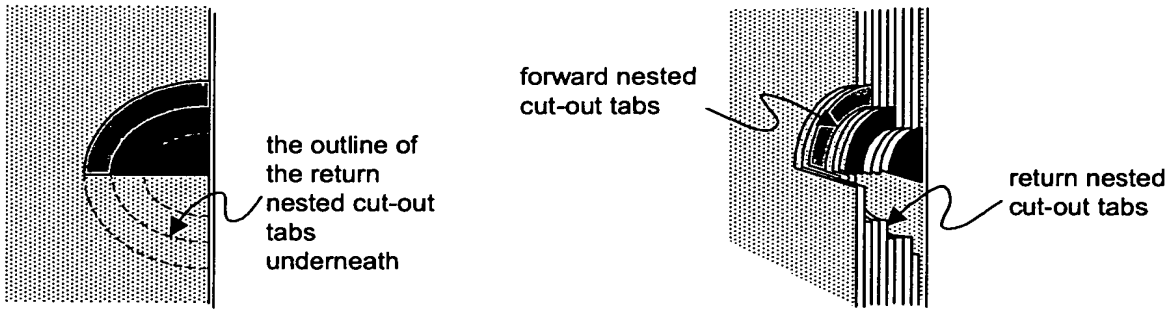


Fig. 10. Merging a forward nested cut-out tabs with its nested return tabs.

The three unbounded edges of the printed document provide three natural dimensions for laying out hyperlinks of distinct functions or concepts. The orthogonal axes of two adjacent edges can readily be mapped to two distinct ways of accessing the content. For instance, in a restaurants guide, one axis may be used for accessing restaurants by location and the other axis by cuisine. In the project plan of Figure 11, the left/right edge is used for accessing the descriptions of the various products and the bottom edge is used for accessing the development plans.

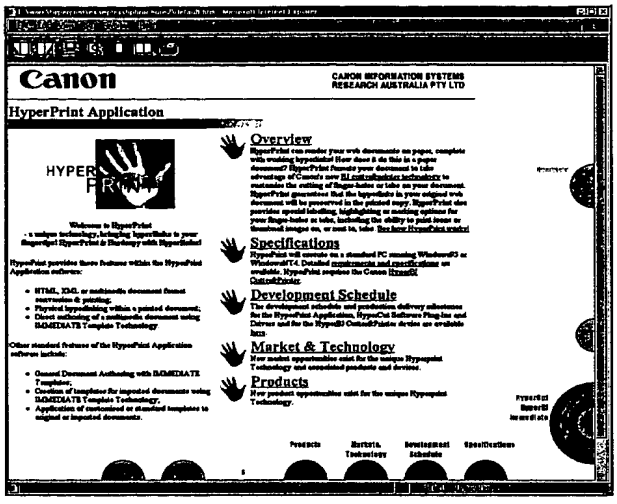


Fig. 11. The screen output of a project plan in print preview mode.

2.5. Printing and cutting

2.5.1. Duplex printing

For a language such as English that reads from left to right, the right edge of the printed hypermedia document will be used for forward links and the left edge for backward links. Backward linking to a right-hand page has to use a cut-out tab that ends at the opposite left-hand page. If the document is printed double-side, then forward linking to a left-hand page also has to be done through a cut-out tab that ends at the opposite right-hand page (Figure 12).

There are several implications. First of all, to support backward links, duplex printing is required (for printing the associated cut-out tabs) even if the content is printed only on one side. Secondly, when referring to the "start/end page" of a tab, we actually imply the start/end page and its opposite page. If the numbering of the pages is known before the optimization process, the conditions for merging the tabs should be relaxed accordingly. Thirdly, if the content is printed on both sides, a clearer means of associating cut-out tabs with hot-spots is especially important as a cut-out tab may become associated with content on the opposite page.

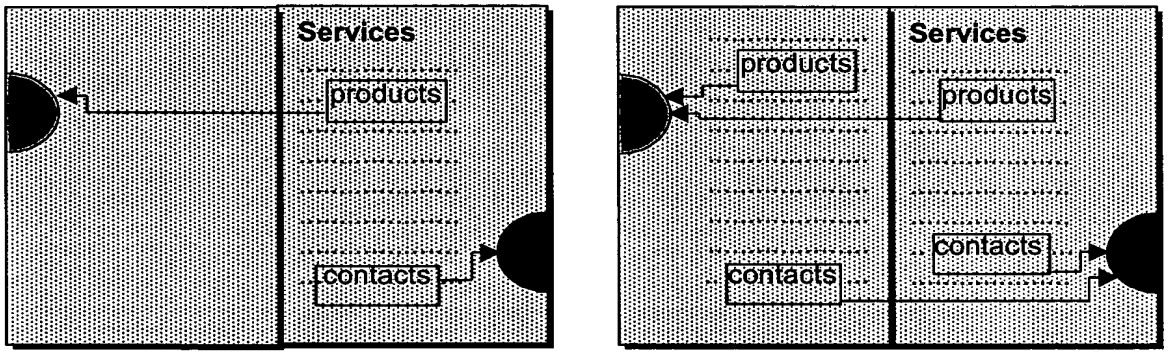


Fig. 12. Duplex printing is required to support backward links. Clear association of cut-out tabs and hot-spots is especially important when the content is printed on both sides.

2.5.2. Paper alignment and col r repr duction

As a cut-out tab may pass through a number of pages, very precise paper alignment is required of the printer/cutter. If the hot-spot and cut-out tabs are color coded, the color printer used must be capable of reproducing the color faithfully so that the set of colors carefully selected on the screen do not become indistinguishable on paper.

. Cut-out tab model

A physical hyperlink from a hot-spot on page S to a target on page T can be represented by a data structure comprising of the 1-dimensional vector $\overrightarrow{ST}$ and the location of the target page, that is, the page number of T. Since the cut-out tabs that implement the hyperlink can only be in either a forward or a backward direction, only bi-dimensional vectors are involved. The same basic data structure is used to represent the cut-out tabs each of which may be associated with multiple hyperlinks. Hereafter, depending on the context, the notation ST is used to denote a hyperlink or a cut-out tab from page S to page T.

Additional attributes such as the location of the tabs on an edge, the shape of the tabs, the colours and labels used by the tabs, etc. are required when rendering the tabs. Default values or author's inputs can be used for the value of those attributes that are not set by the optimization process. The author should be allowed to examine the results of the optimization process and make adjustments as required.

As mentioned earlier, a cut-out tab running from S to T has to pass through all the pages in between and is visible - although not necessary required by - those pages. Appropriate presentational style will be used to distinguish an inactive tab from an active one, for instance by removing or greying out the label of a tab when it has no associated hyperlink. An array must be added to the data structure of the cut-out tab to store the status of the tab for the intermediate pages. In case the tab is shared by multiple hyperlinks, a second array is used to store the labels/icons for the intermediate pages.

4. The optimization process

4.1. Overviews

The optimization process starts with a set of cut-out tabs each representing one hyperlink to be implemented physically. The set of cut-out tabs is then gradually reduced by combining cut-out tabs which can serve multiple hyperlinks.

Other physical attributes such as the size of the cut-out tabs and the page as well as certain system parameters such as the edges to be used for tabs, the maximum number of tabs that can be nested along a single axis and the density of the tabs along an edge are required when "optimizing" (in a perceptual sense) the placement of the cut-out tabs.

4.2. Graphical Notation

The graphical notations of Figure 13 will be used to depict the cut-out tabs diagrammatically. The graphical notations can also be used in the graphical user interface (GUI) of an authoring system or editing tool to visualise the assignment and the physical arrangement of the tabs.

4.3. Optimizing the number of tabs

The optimization process involves the following basic steps:

1. Define the set of hyperlinks that will be implemented with cut-out tabs.
2. Assign a cut-out tab to each hyperlink.
3. Reduce the number of tabs by sharing the tabs among the hyperlinks.
4. Reduce the number of axes by nesting tabs.
5. Reduce the number of axes by aligning non-overlapping tabs.

The optimization steps can be automated. However, editing tools that allow the author to adjust the resulting hyperlinks/cut-out tabs assignment interactively are desirable.

In the next section, we will describe the optimization process in greater details using a restaurants guide example.

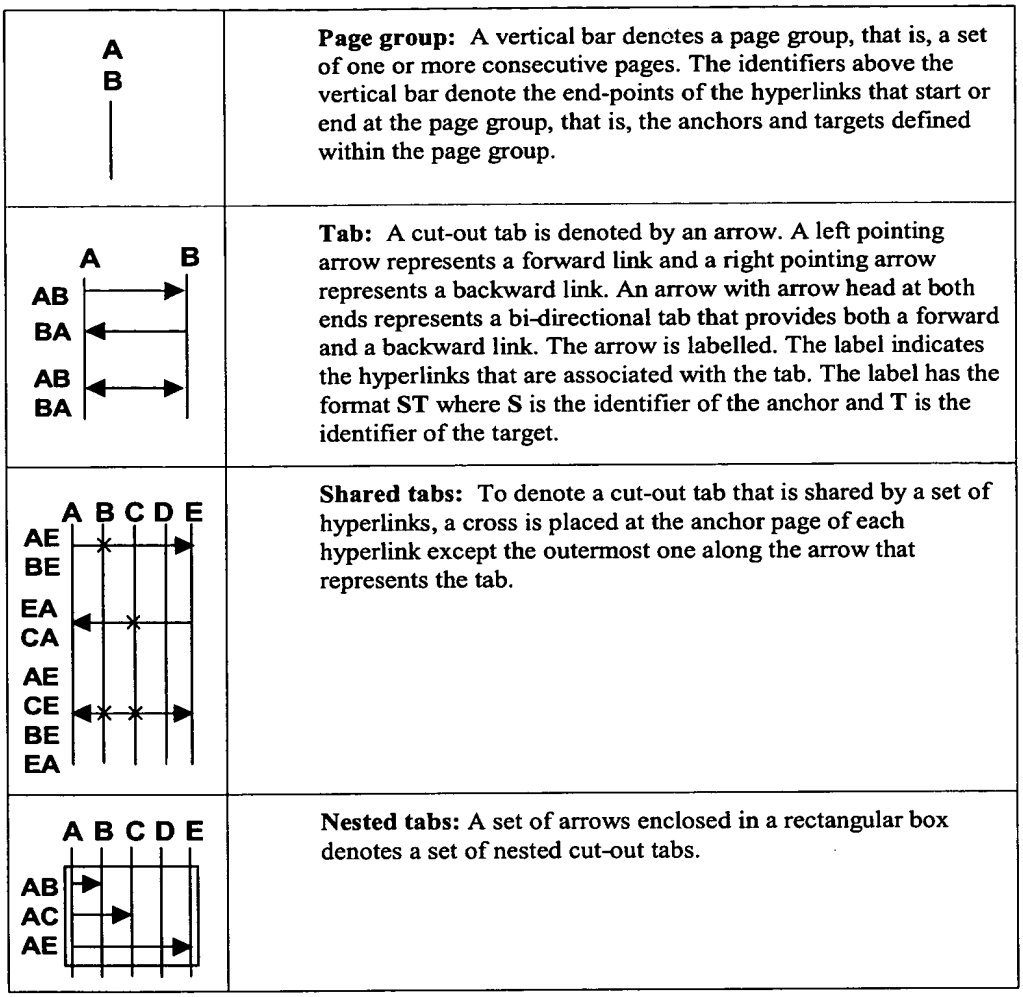


Figure 13: Graphical notations of cut-out tabs.

5. A restaurants guide example

Consider a restaurants guide that lists restaurants in a number of suburbs according to their specialty. The root (or starting) page of the restaurant guide is denoted as **R**, followed by the restaurant directories for m types of cuisines $\mathbf{D}_i, i = 1,\ldots,m$. Then come the suburb sections, one for each of the n listed suburbs. Each suburb section has an overview page $\mathbf{S}_j$ and m cuisine sub-sections $\mathbf{C}_{ij}, i = 1,\ldots,m$, one for each of the m cuisines where $1 \le j \le n$. The cuisine sub-sections contain the restaurants' details. In summary, the document has the structure:

$$\mathbf{R}\ \mathbf{D}_1\ \mathbf{D}_2 \ldots \mathbf{D}_m\ \mathbf{S}_1\ \mathbf{C}_{11}\ \mathbf{C}_{21} \ldots \mathbf{C}_{m1} \ldots \mathbf{S}_n\ \mathbf{C}_{1n}\ \mathbf{C}_{2n} \ldots \mathbf{C}_{mn}$$

5.1. Defining the hyperlinks

The hyperlinks defined are listed in Table 1. Each of the hyperlinks listed in Table 1 is assigned a cut-out tab. Additional hyperlinks that link each directory entry to the restaurant may be defined for the display version.

The set of cut-out tabs is initially given by:

$$\mathbf{H}_0 = \bigcup_{i=1}^{m}\{\mathbf{RD}_i\} \cup \bigcup_{r=1}^{m}\bigcup_{\substack{s=1 \\ s\neq r}}^{m}\{\mathbf{D}_r\mathbf{D}_s\} \cup \bigcup_{i=1}^{m}\bigcup_{j=1}^{n}\{\mathbf{D}_i\mathbf{S}_j\}$$
$$\cup \bigcup_{j=1}^{n}\bigcup_{i=1}^{m}\{\mathbf{S}_j\mathbf{D}_i\} \cup \bigcup_{j=1}^{n}\{\mathbf{RS}_j\} \cup \bigcup_{j=1}^{n}\bigcup_{i=1}^{m}\{\mathbf{S}_j\mathbf{C}_{ij}\}$$
$$\cup \bigcup_{j=1}^{n}\bigcup_{r=1}^{m}\bigcup_{\substack{s=1 \\ s\neq r}}^{m}\{\mathbf{C}_{rj}\mathbf{C}_{sj}\} \cup \bigcup_{j=1}^{n}\bigcup_{i=1}^{m}\bigcup_{r=1}^{m}\{\mathbf{C}_{rj}\mathbf{D}_i\}$$

- a total of $n(1+2m+2m^2)+m^2$ tabs.

Repetitive structures and simple hyperlinks linking various sections of the restaurants guide are used in the example to make the optimization process easier to understand and its effectiveness easier to demonstrate. The optimization method can be applied to documents with arbitrary cross reference hyperlinks although the physical constraints of the printed page might prevent some of the optimized cut-out tabs from being implemented.

5.2. Sharing cut-out tabs

A cut-out tab can be used for multiple hyperlinks that end at the same page as the tab. In fact, a tab **AB** can potentially be used for any hyperlink that starts at **A** or at a section between **A** and **B** and ends at **B** provided that the tab is properly labelled on all the pages it passes through and that such use is not confusing to the reader. For instance, referring to the final tab assignment in Figure 14, the tab $\mathbf{RD}_m$ running from **R** to $\mathbf{D}_m$ is shared by hyperlinks $\mathbf{RD}_m$ and $\mathbf{D}_r\mathbf{D}_m$ where r=1,...,m-1. In addition, a tab **AB** leading from a page **A** to a page **B** provides an implicit return path from page **B** to page **A**. Hence, the number of tabs can be reduced by merging tab pairs that have their two tabs start on the page the other tab ends.

By sharing tabs among hyperlinks, the original set of cut-out tabs is reduced to:

$$\mathbf{H}_1 = \bigcup_{i=1}^{m}\{\mathbf{RD}_i\} \cup \bigcup_{s=1}^{m}\{\mathbf{C}_{mn}\mathbf{D}_s\} \cup \bigcup_{j=1}^{n}\{\mathbf{RS}_j\}$$
$$\cup \bigcup_{j=1}^{n}\bigcup_{s=1}^{m}\{\mathbf{S}_j\mathbf{C}_{sj}\} \cup \bigcup_{j=1}^{n}\bigcup_{s=1}^{m-1}\{\mathbf{C}_{mj}\mathbf{C}_{sj}\}$$

The total number of tabs is now $2nm + 2m$.

Table 1: Hyperlinks in the Restaurants Guide example

Hyperlinks	Description
$\mathbf{RD}_i$, $i = 1,\ldots,m$	from the root page to each cuisine directory
$\mathbf{D}_r\mathbf{D}_s$, $r,s = 1,\ldots,m$, $r \neq s$	from each cuisine directory to another cuisine directory
$\mathbf{D}_i\mathbf{S}_j$, $i = 1,\ldots,m$, $j = 1,\ldots,n$	from each cuisine directory to each suburb section
$\mathbf{S}_j\mathbf{D}_i$, $i = 1,\ldots,m$, $j = 1,\ldots,n$	from each suburb section to each cuisine directory
$\mathbf{RS}_j$, $j = 1,\ldots,n$	from the root page to each suburb section
$\mathbf{S}_j\mathbf{C}_{ij}$, $i = 1,\ldots,m$, $j = 1,\ldots,n$	from each suburb section to each of its local cuisine sections
$\mathbf{C}_{rj}\mathbf{C}_{sj}$, $r,s = 1,\ldots,m$, $r \neq s$, $j = 1,\ldots,n$	from each cuisine section of a suburb to the other cuisine sections of the same suburb
$\mathbf{C}_{rj}\mathbf{D}_i$, $r,i = 1,\ldots,m$, $j = 1,\ldots,n$	from each cuisine section of a suburb to the cuisine directories

3. Nesting cut-out tabs

Related tabs are nested to reduce the total edge space required by the tabs. A set of nested tabs should link materials that have a natural or logical relationship. Such relationship may be inferred from the structure of the document and other information provided by the author. For instance, the set of nested tabs $\mathbf{RD}_m$ in Figure 14 are used for accessing the m restaurant directories each for a different cuisine. In this case, nesting the tabs reduces the number of axes required from m to 1.

The innermost unions of each of the terms of $\mathbf{H}_1$ can be nested. Assuming that the number of suburbs n and the number of cusine m are small enough that nesting n and m cut-out tabs is visually acceptable, after nesting the tabs, the total number of (nested) tabs becomes $2n+3$.

5.4. Aligning axes

The last two terms of $\mathbf{H}_1$ contain two sets of non-overlapping (nested) tabs $\bigcup_{s=1}^{m}\{\mathbf{S}_j\mathbf{C}_{sj}\}$, $j=1,\ldots,n$ and $\bigcup_{s=1}^{m-1}\{\mathbf{C}_{mj}\mathbf{C}_{sj}\}$, $j=1,\ldots,n$. Each of the two sets can share a single axis. As a result, the $2n+3$ (nested) tabs require only 5 axes.

The final tab assignment is shown in Figure 14. Figure 15 shows the screenshots of a restaurants guide example. The Web document was created using Hyperprint – a prototype authoring system which we are going to described in the next section. All Hyperprint documents are provided with a tool bar in the top frame to allow the user to flip through the document and switch between display and print preview modes.

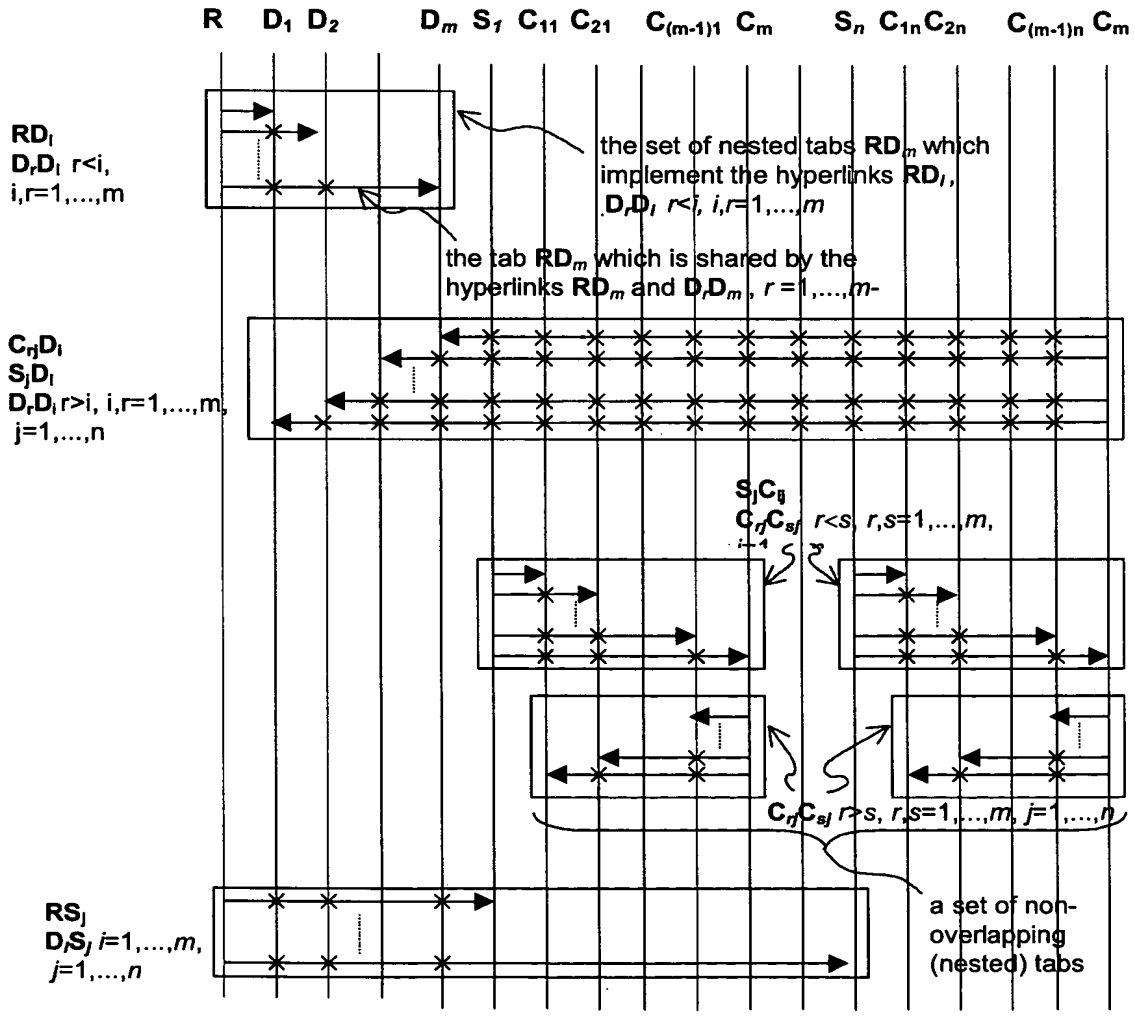
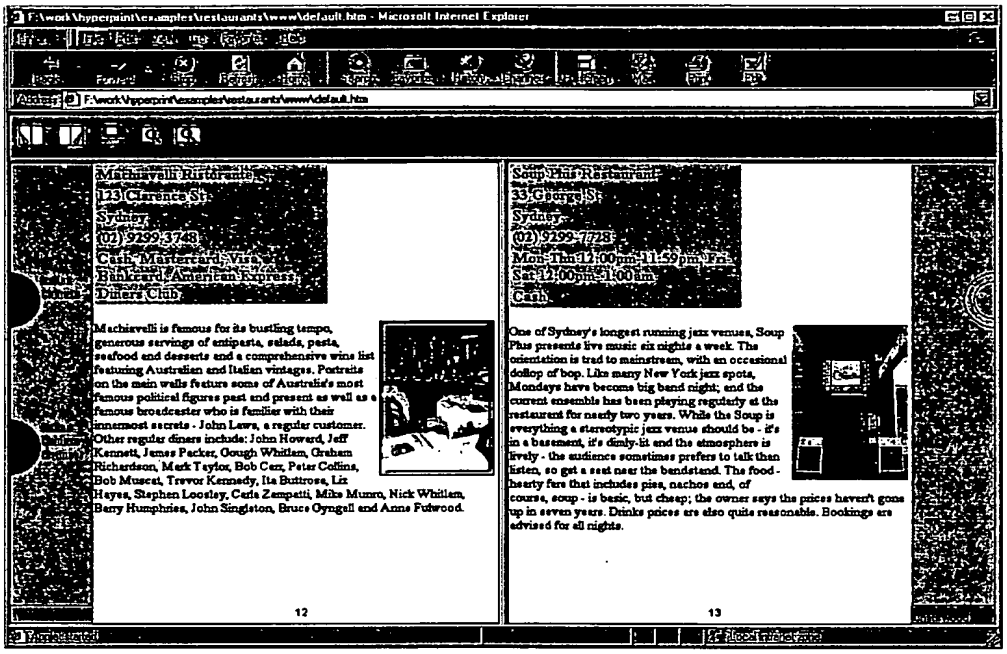
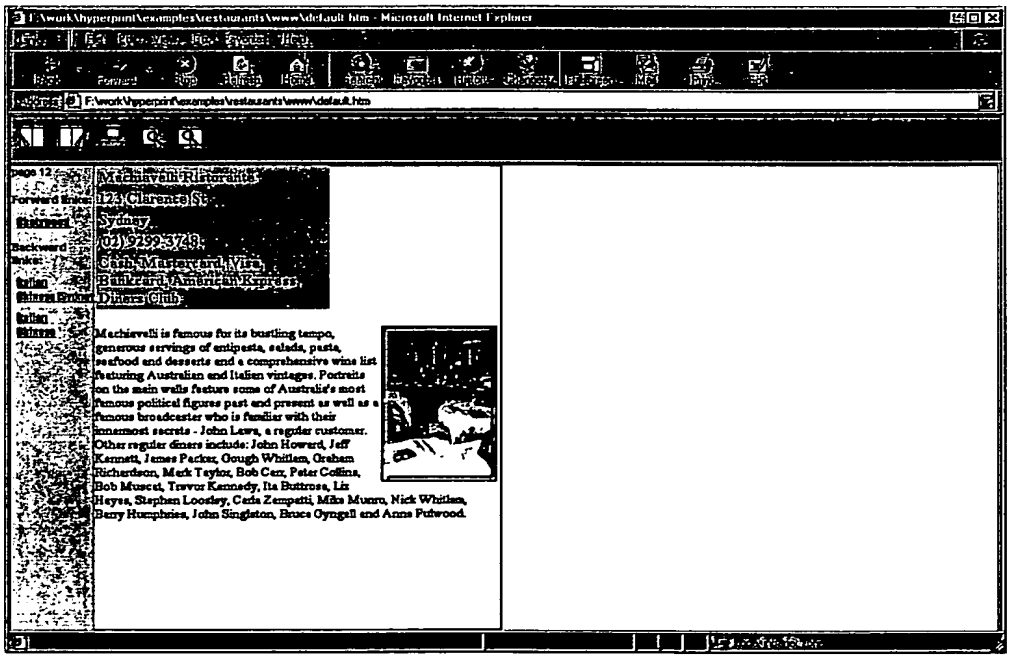


Fig. 14. The final tab assignment for the Restaurants Guide example.



(a)



(b)

Fig. 15. Screenshots of the restaurants guide example in (a) print preview mode, and (b) display mode.

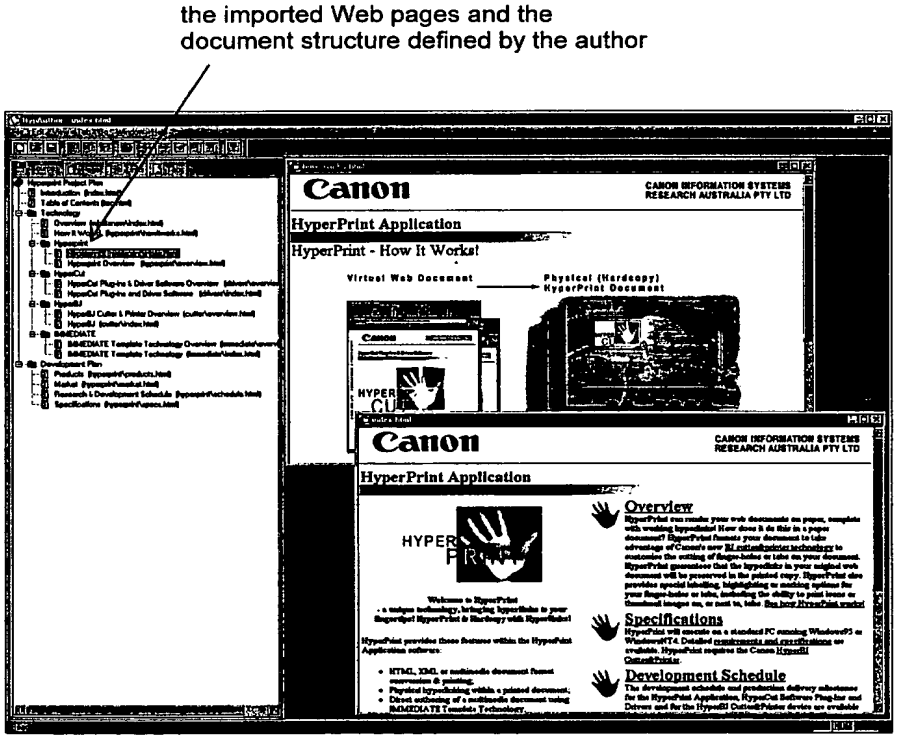
A12

Hyperprint

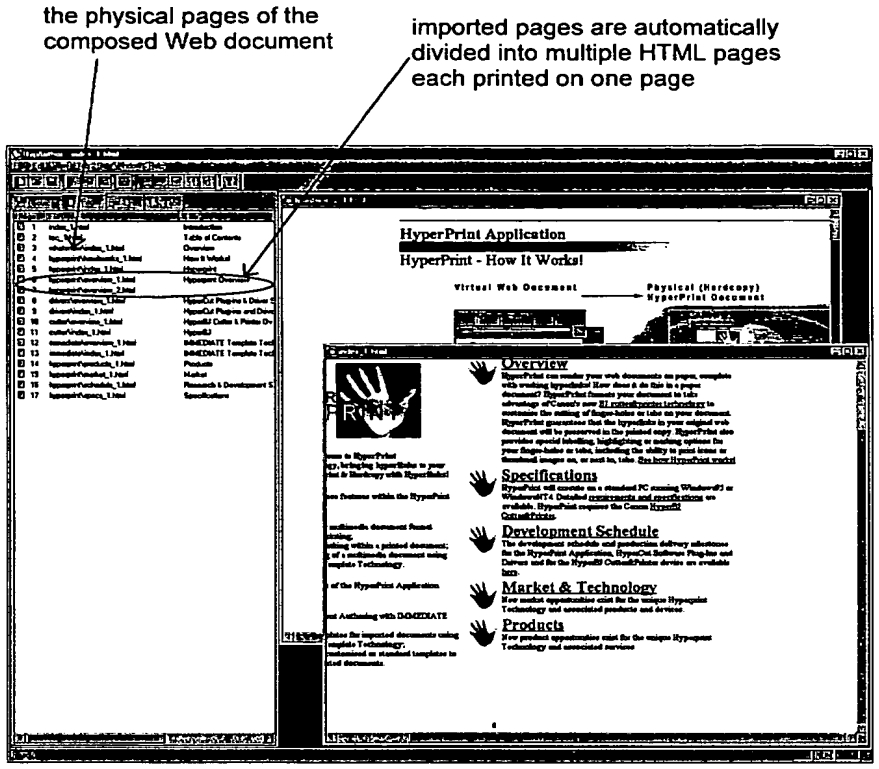
An authoring system called Hyperprint is being developed. The system creates Web documents that can be printed and cut to support hyperlinks in their printed copies. It implements the optimization procedure we outlined in the previous section. The first prototype of the system allows the author to:

1. import a set of HTML (HyperText Markup Language) Web pages to a document - the original HTML pages do not need to be organized in pages of similar length.
2. divide an imported HTML page into multiple HTML pages each printed on one physical page - this is done automatically at the time the HTML page is imported
3. specify the tree structure of the document
4. select the hyperlinks to be physically implemented
5. optimize the hyperlinks/cut-out tab assignment
6. re-position the tabs and specify their styles
7. save the final HTML pages

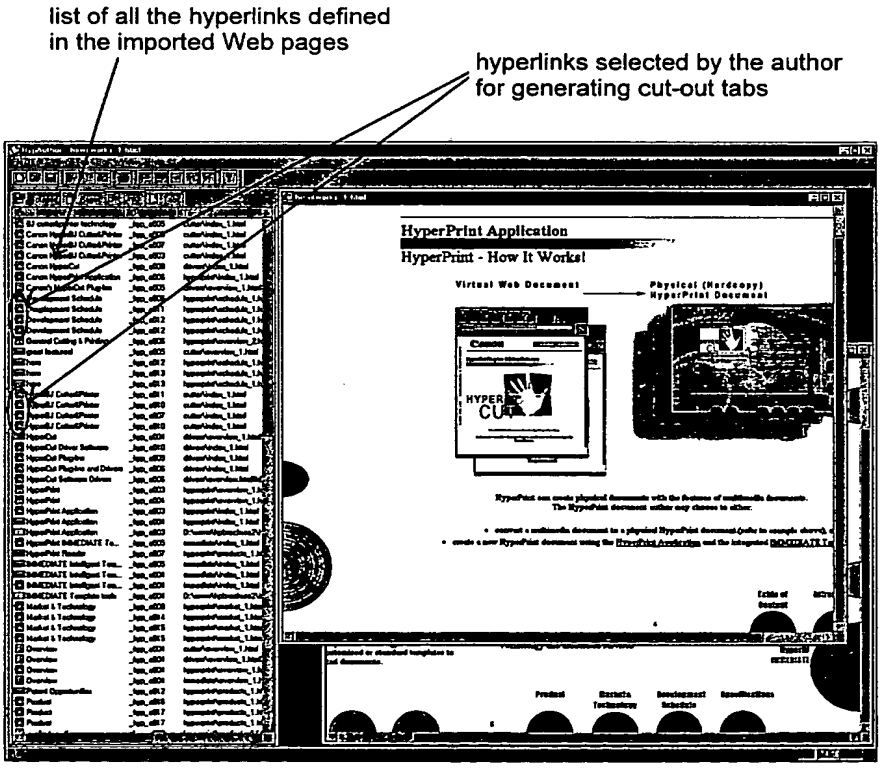
Figure 16 shows the user interface of Hyperprint and the main steps of composing a Web document with Hyperprint using a project plan as an example. Figure 17 shows the printed (and cut) copy of the project plan produced. Unlike the restaurants guide example, the project plan does not have repetitive structures and has arbitrary cross reference hyperlinks.



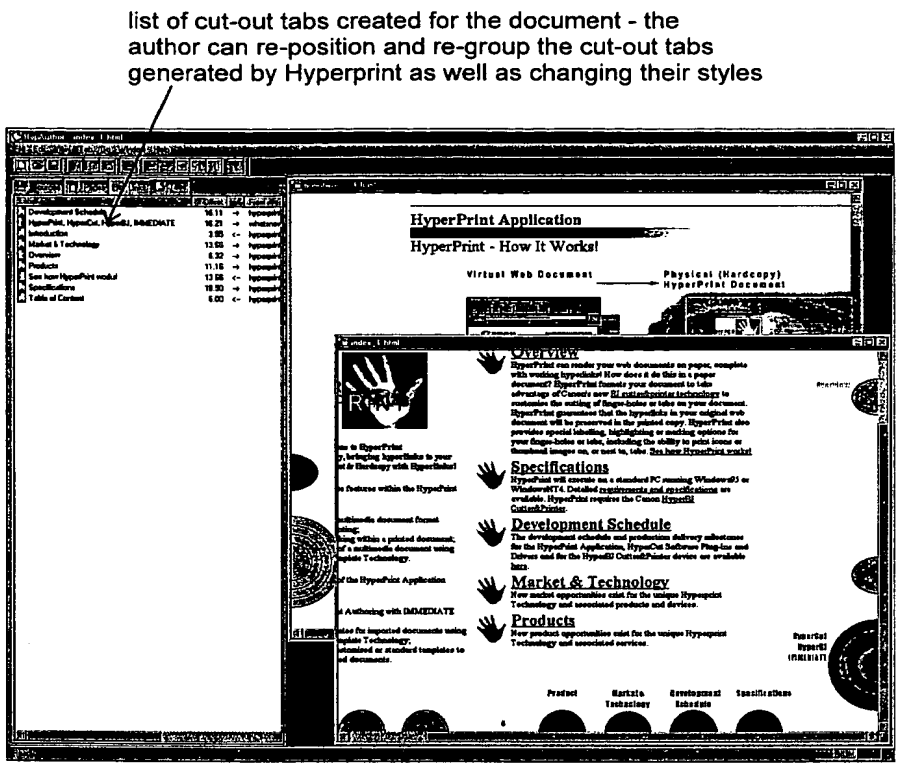
(a)


the physical pages of the composed Web document
imported pages are automatically divided into multiple HTML pages each printed on one page
HyperPrint Application
HyperPrint - How It Works!
Virtual Web Document
Physical (Hardcopy) HyperPrint Document
Overview
Specifications
Development Schedule
Market & Technology
Products

(b)


list of all the hyperlinks defined in the imported Web pages
hyperlinks selected by the author for generating cut-out tabs
HyperPrint Application
HyperPrint - How It Works!
Virtual Web Document
Physical (Hardcopy) HyperPrint Document

(c)



(d)

Fig. 16. The user interface of Hyperprint and the main steps in composing a Hyperprint document from a set of Web pages: (a) importing the source pages and defining the document structure, (b) dividing the source pages into multiple HTML files each printed on one physical page, (c) selecting the hyperlinks to be implemented as cut-out tabs and (d) generating the cut-out tabs and defining their styles.

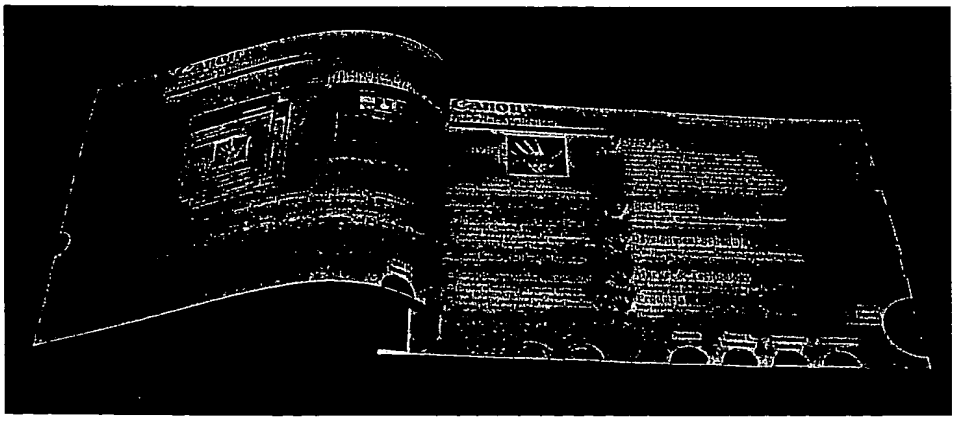


Fig. 17. The printed (and cut) copy of a project plan created with Hyperprint.

The generated HTML pages can be displayed by browsers such as the Internet Explorer 4 (IE4) that support CSS1 [2] and the Document Object Model (DOM) [3]. The HTML pages use applets for rendering the cut-out tabs. The cut-out tabs are functional both on screen and on paper. On screen, clicking the tab causes the target page to be displayed.

The parameters of the cut-out tabs are stored as Javascript variables in a "parameter" script. The "parameter" script is imported into each of the HTML pages together with a set of library scripts. Among other things, the library scripts provide a set of functions that the HTML pages may call to create HTML elements for the cut-out tabs.

can be used for printing the HTML pages of the document. At the moment, a graphic design package is used to drive cutter and produce the cut-out tabs. The cut pages are then wire bound. A utility is being developed for printing and cutting the documents using a printer/cutter.

7. Discussion

Our method allows Web documents and other hypermedia documents to be published in printed copies exploiting both the advantages of the print media and a hyperlinked form. We feel that it is most useful for documents that possess some of the following properties:

- the document is well-structured e.g. a technical report
- the document has a well-defined scope and function e.g. an operation manual
- the document contains related material that can be read in different orders e.g. a reference manual
- the document contains related material that provides multiple views e.g. a maintenance manual that is used by technicians specialized in different areas
- the document contains material that is suitable for content-based exploration
- the document contains material that is mostly locally linked, that is, most of the cut-out tabs required are shallow cut-out tabs that can share their axes rather than deep cut-out tabs that cut through a large number of unrelated pages
- the document contains material that requires frequent cross-reference to content on different pages e.g. a financial report
- the document contains personalized and frequently updated content e.g. online subscriptions and program guides; the particular preference and interest of the individual helps to limit the number of hyperlinks required
- hardcopy is an essential or the preferred recording/communication media e.g. business plans.

At this stage, we have concentrated on developing tools for Web documents using specific examples. Our experience so far have revealed a number of practical issues:

1. The printer used must be capable of accurately aligned duplex printing so that the tabs can be cut precisely.
2. The printer used must be able to produce reasonably accurate color as the cut-out tabs are usually color coded.
3. Reasonably thick paper is required so that the pages are easier to flip and the cut edges are more durable. Nevertheless, we found that the thicker paper used for color printing is often adequate for the purpose.
4. The printed copy has to be bound properly for the cut-out tabs to be usable. However, we found that even common folders that use locking rings produce quite good results.
5. A portion of the edge space should be reserved and not used for cut-out tabs so that the ease of turning pages in paper documents is preserved.

8. Conclusions

We have described a method that allows a hypermedia document to retain its hyperlinks in the printed copy. The method associates the hyperlinks with cut-out tabs on the edges of the printed pages. We discuss several presentational styles for the cut-out and their associated properties. A method for modelling the cut-out tabs and optimising their assignment to the hyperlinks was also presented. We also briefly described a prototype authoring system, Hyperprint, that we have built. The system allows an author to construct Web documents that can be printed and cut to support hyperlinks in their printed copies. The more general problem of representing dynamic content such as audio, video and animation of multimedia document on paper media requires further research.

Defying the critics, paper documents have continued to survive and proliferate in the electronics age. Ted Nelson has described today's hypermedia as "tangled and limited", and that it has "no deep structure" and has "brought us interconnection but not understanding" [1]. In contrast, traditional paper documents usually have a well-defined structure and one expository sequence. Our method requires the author to organize his/her hypermedia document in the same way as traditional paper document and not just interconnecting content. It also requires the author to put in some thought as to which hyperlinks are more important to the readers and, hence, should be implemented as cut-out tabs in the printed copy. However, the additional effort will lead to easier and better reading and should not be seen as a burden. Moreover, authoring tools such as Hyperprint can greatly reduce the effort involved.

We have not conducted usability trials for different applications. Areas that we feel have great potential include:

- marketing brochures of products/services,
- program guides,
- online subscriptions,
- information brochures such as facilities guide, restaurants guide, tourist guide, etc.,
- project/business plans and reports,
- manuals, and
- children books.

Documents with a standard structure, such as project plans and business reports, allow template based authoring which simplifies layout optimization.

References

[1] Nelson, T. (1997). *The Future of Information*. Limited distribution by ASCII Corporation, Tokyo. Available at http://www.sfc.keio.ac.jp/~ted/INFUTscans/INFUTscans.html.

[2] *Cascading Style Sheets Level 2 Specification*. Available at http://www.w3.org/TR/REC-CSS2/.

[3] *Cascading Style Sheets Level 1 Specification*. Available at http://www.w3.org/TR/REC-CSS1-961217.html.

[4] *Document Object Model Level 1 Specification*. Available at http://www.w3.org/TR/PR-DOM-Level-1/.

[5] Waters, S., Butler, S. and MacMillan, R. *Internet and Intranet Printing*. Available at http://www.w3.org/Printing/iiprnt.html.

Vitae

Ernest Wan has a Ph.D. in Computer Science from the Australian National University. He is a Research Staff Member at Canon Information Systems Research Australia. His research interests include neural networks, knowledge-base and database systems, and internet-related technologies.

Philip K. Robertson is General Manager of Research and Development at Canon Information Systems Research Australia. He has a PhD in Computer Science from the Australian National University, and has published widely in image processing, graphics, color and visualization. He is on the editorial board of IEEE TVCG, IEEE CG&A, and Computers and Graphics, and has served on program committees for international conferences in these fields.

John Brook graduated from Canterbury University, New Zealand in 1983. He is currently working for Canon Information Systems Research Australia. His research interests include video and multimedia systems and applications, and internet-related activities.

Stephen Bruce received his B.D. in 1992 from the University of Technology Sydney in Visual Communication. Currently he is working as a Graphic Designer at Canon Information Systems Research Australia.

Kristine Armitage received her BA in Visual Arts degree from the University of Western Sydney, Nepean, N.S.W. in 1994 and at the same institution an Associate Diploma in Graphic Design and Photography in 1996. She has been part of the art department of Canon Information Systems Research Australia for three years designing promotional material and artwork for software developed by the company.